Government of **Western Australia**
School Curriculum and Standards Authority

SAMPLE TEACHING AND LEARNING OUTLINE

TECHNOLOGIES

DIGITAL TECHNOLOGIES

YEAR 6

2020/22074v2 [PDF 2020/33387]

This document is an introduction to planning a teaching and learning outline with syllabus content for Year 6 Digital Technologies. It provides suggested sequencing and timing for teaching the syllabus content. For further details on curriculum requirements and available options, teachers should refer to the School Curriculum and Standards Authority's (the Authority's):

- *Policy Standards for Pre-primary to Year 10: Teaching, Assessing and Reporting*
- Table 1: *Western Australian Curriculum and Assessment Outline*: curriculum requirements and available options.

Schools may choose to teach the syllabus content for two hours per week for a semester, **or** one hour per week for the year. Sample plans provide a range of possible learning experiences from which assessment should be drawn. This *Year 6 Sample Teaching and Learning Outline* provides teachers with possible learning experiences over 18 weeks (two hours per week) and unpacks the syllabus content to assist teachers in their understanding. Teachers may choose from the **Suggested teaching and learning experiences** that suit the context of the school.

A presentation (*Western Australian Curriculum Technologies Presentation*), which unpacks the process to develop this plan, is available on the Presentations page of the Authority website (https://k10outline.scsa.wa.edu.au/home/resources/presentations).

Year 6 Syllabus Content – Digital Technologies

Content	Description
Digital systems	Digital systems have components with basic functions and interactions that may be connected together to form networks which transmit different types of data
Representation of data	Whole numbers are used to represent data in a digital system
Collecting, managing and analysing data	Collect, sort, interpret and visually present different types of data using software to manipulate data for a range of purposes
Digital implementation	Design, modify, follow and represent both diagrammatically, and in written text, simple algorithms (sequence of steps) involving branching (decisions) and iteration (repetition) Implement and use simple visual programming environments that include branching (decisions), iteration (repetition) and user input Manage the creation and communication of information, including online collaborative projects, using agreed social, ethical and technical protocols
Investigating and defining	Define a problem, and a set of sequenced steps, with users making decisions to create a solution for a given task Identify available resources
Designing	Design, modify, follow and represent both diagrammatically, and in written text, alternative solutions using a range of techniques, appropriate technical terms and technology
Producing and implementing	Select, and apply, safe procedures when using a variety of components and equipment to make solutions
Evaluating	Develop collaborative criteria to evaluate and justify design processes and solutions
Collaborating and managing	Work independently, or collaboratively when required, considering resources, to plan, develop and communicate ideas and information for solutions

Year Level Description

In Year 6, students further develop understanding and skills in computational thinking such as identifying similarities in different problems and describing smaller components of complex systems. They will have opportunities to create a range of solutions, such as quizzes and interactive stories and animations that involves more than one branching solution (choice of options).

Students consolidate their understanding of the role individual components of digital systems play in the processing and representation of data. They acquire, validate, interpret, track and manage various types of data, and begin to explain the concept of data states in digital systems and how data are transferred between systems.

Students learn to further develop abstractions by identifying common elements across similar problems and systems and make connections between models and the real-world systems they represent.

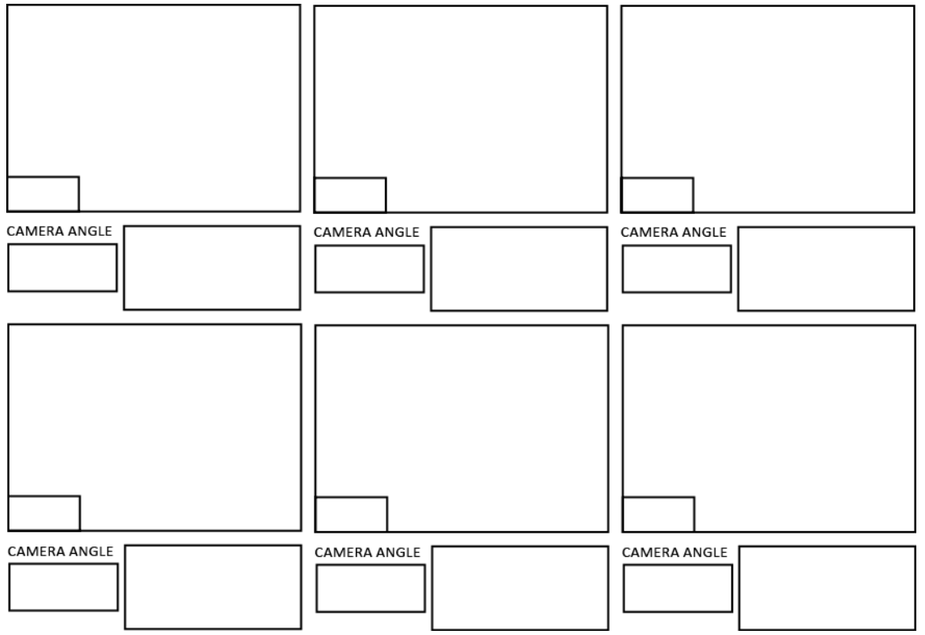
When creating solutions, students further refine their skills to identify and use appropriate data and requirements. They increase the sophistication of their algorithms by identifying repetition. They learn to incorporate repeat instructions or structures when implementing their solutions through visual programming environments, such as reading user input until an answer is guessed correctly in a quiz.

Students critique design solutions and examine the sustainability of their own, and existing, information systems.

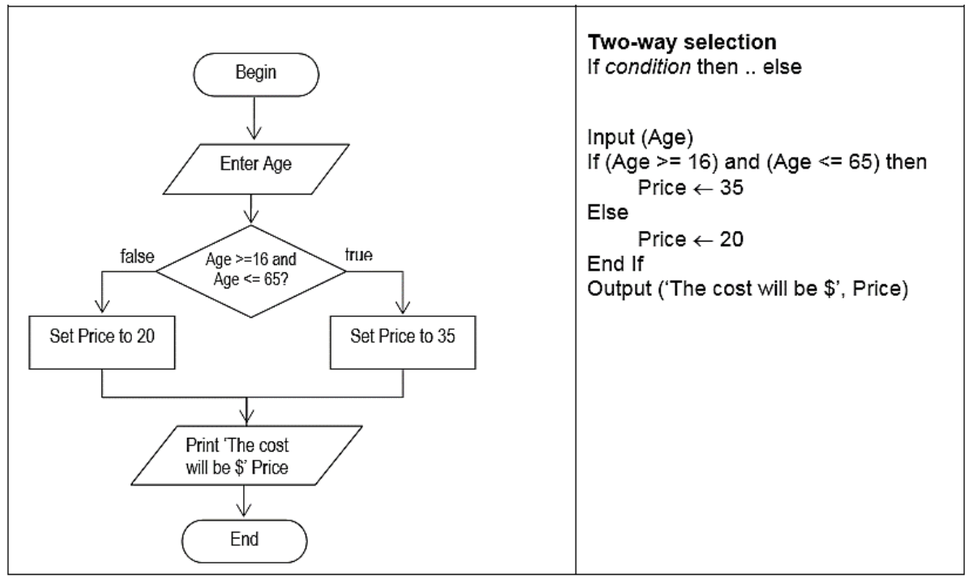
Students develop strategies to communicate information and ideas using agreed social, ethical and technical protocols, taking into account the safety aspects of working in digital environments.

Year 6 Learning Area: Technologies – Digital Technologies			
Year 6 Achievement Standard At Standard, students outline interactions between components and basic functions within digital systems and how they transmit different types of data to form networks. They make a connection between whole numbers being used to represent data within a digital system. They use software to collect, sort, interpret, visually present and manipulate data for a range of purposes. Students use simple visual programming environments to design, modify, follow and represent both diagrammatically, and in written text, algorithms (sequence of steps), involving branching (decisions), iteration (repetition) and consider user input. Students manage, create and communicate information for online collaborative projects, using agreed social, ethical and technical protocols. In Digital Technologies, students identify available resources to design a solution for a given digital task, outlining problem-solving decisions, using algorithms (sequenced steps). Students develop alternative solutions by designing, modifying and following both diagrammatically and in written text, using a range of appropriate technical terms, technologies and techniques. They select and apply safe procedures when using a variety of components and equipment to make solutions. Students develop criteria collaboratively to evaluate and justify design processes and solutions. They work independently, or collaboratively, considering resources and safety to plan, develop and communicate ideas and information for solutions.			
Weeks	**Syllabus content**	**Content unpacked**	**Suggested teaching and learning experiences**
1–2	**Producing and implementing** Select, and apply, safe procedures when using a variety of components and equipment to make solutions	• When producing solutions, students are required to do so safely. Safe use of hardware and software should represent the school's Information Communication Technologies (ICT) code of conduct and the classroom rules set by the teacher. Provide a copy of the ICT code and explain consequences of breaches. • Social and ethical protocols when working collaboratively online; for example: ▪ suitability of content, e.g. acceptable language, originality of work (copyright and plagiarism) ▪ respecting input from all group members ▪ not publishing personal identifying information of self or others (e.g. names, school, images) ▪ adhering to school's ICT code of conduct.	• Students create a set of rules that mimic an ICT code of conduct. This should cover hardware and software safety, and its correct use in and out of the classroom. After the class creates the list of rules, students individually rank them based on their dangers. This data is then used for a class discussion. • Students can create posters or infographics for Year 1 students showing a breach or the consequences of breaching the school's ICT policy. Students can then deliver them to the Year 1 classroom with a brief explanation. • Use resources provided by the eSaftey Commissioner classroom resources (https://www.esafety.gov.au/educators/classroom-resources) to help students understand online safety, including: ▪ Cybersmart challenge ▪ The Lost Summer ▪ Making Good choices online ▪ #Game on. These resources can be used in or out of the classroom.
3–5	**Digital systems** Digital systems have components with basic functions and interactions that may be connected together to form networks which transmit different types of data **Evaluating** Develop collaborative criteria to evaluate and justify design processes and solutions	• All systems involve an input, a process and an output. • Digital systems involve inputting data through a range of peripherals, processing through computer hardware, such as Central Processing Units and primary storage, and outputting information or data through output devices. • Peripherals are non-essential to the running of a device or computer system, but increases useability for the end user. • Distinguish between hardware and peripherals. • Input devices include keyboards, mice, barcode scanners, game controllers, scanners etc. • Output devices include monitors, printers, speakers etc. • These devices can be wired or wireless. Data needs to be transmitted to these devices to receive data. • A network refers to more than one device connecting and communicating together. • Peer-to-peer networks – no central server, with devices having the same privileges and access as each other. • Client-server networks – a server attached to the network and other devices can share the server's resources. The server sets the restrictions and protocols that are to be used on the network. • Network Interface Cards (NICs) are devices that communicate on a network, allowing for devices to connect to a network. These are either wireless or wired. A device cannot connect to a network without a NIC.	• Classroom discussion: ▪ What computer devices do student have access to at school? ▪ What computer devices do students have access to at home? • Students work in pairs to: ▪ create a list of all the components of a computer system/s they have at home or in the classroom and describe the basic function of each ▪ classify each component as an input or output device ▪ use a collaborative online tool to record their responses, e.g. Padlet™, Google Docs™, OneNote™. • View one of the following videos about wireless communication: ▪ How WiFi and Cell Phones Work | Wireless Communication Explained (https://youtu.be/kxLcwIMYmr0) ▪ How does wireless work? (https://www.grandmetric.com/2018/03/01/explained-how-does-wireless-work/). • Discuss the types of networks that students commonly use at school and at home for different devices. • Students work in pairs to create a hyperlinked PowerPoint® quiz, including questions relating to describing digital systems content. PowerPoints will then be shared online (e.g. via the school's learning management system, class wiki etc.) for others to access. ▪ Discuss what a hyperlink is and demonstrate how to create hyperlinks in PowerPoint. Refer to Add a hyperlink to a slide (https://support.office.com/en-us/article/Add-a-hyperlink-to-a-slide-239C6C94-D52F-480C-99AE-8B0ACF7DF6D9). ▪ Create the PowerPoint quiz on any digital systems topics previously delivered. These need to be non-linear in nature, with branching (decisions) providing feedback to the user on their responses as correct or incorrect. ▪ Upload the PowerPoint quiz to the online collaborative site (determined by the teacher). ▪ Feedback can be given in the form of peer evaluation after creating agreed criteria for evaluation.

Weeks	Syllabus content	Content unpacked	Suggested teaching and learning experiences
		• Define hyperlinks and explain their purpose, i.e. a link that is used to take the user to another location. • Different formats or ways that hyperlinks can be inserted: ▪ text ▪ images ▪ buttons. • Elements or content that should be included in a flowchart or storyboard. • Structure of the quiz; for example: ▪ give clear instructions to the user ▪ use hyperlinks to facilitate the implementation of the quiz ▪ include questions, such as multiple-choice, true/false, selection of the correct image/term.	• Using their choice of software, students create an infographic that describes the advantages and disadvantages of the different ways to transmit data between computers, i.e. wired vs wireless connections: • Prior to commencing the infographic task, discuss as a class: ▪ available software/apps ▪ target audience ▪ elements of design ▪ possible content to include ▪ copyright considerations. • Students evaluate infographics provided by the teacher to determine the quality of content and effectiveness of different infographics. • To begin the first part of the task, students plan or design an infographic, including brief annotations to describe why they used the design elements implemented. These include colours used, shapes etc. • Students continue working on the infographic task, including: ▪ finalising the plan ▪ choosing between online infographic creations and paint etc. and justifying their decision ▪ researching and selecting relevant images and content ▪ creating, saving and printing infographic using selected software.
6–8	**Representation of data** Whole numbers are used to represent data in a digital system **Designing** Design, modify, follow and represent both diagrammatically, and in written text, alternative solutions using a range of techniques, appropriate technical terms and technology	• Computers can only process, send or receive data in whole numbers. These are referred to as binary, and are represented as either ones (1) or zeros (0). • In binary, electronic pulses are either off (0) or on (1). • All data that is represented will be processed to binary regardless of being text, sound, video, image etc. • Diagrammatic designs include storyboards, flowcharts, site maps etc. All diagrams need written text to inform the purpose or explain details. These can include labels or annotations. • When technical terms are used, an explanation should be given.	• Pixels and binary digits: ▪ Pixels and Binary Digits assessment ideas (https://www.digitaltechnologieshub.edu.au/teachers/assessment/assessment-ideas/pixels-and-binary-digits). • Brainstorm codes that students may be familiar with, e.g. Morse code, barcodes, Quick Response (QR) codes. • Explain that binary is a method used to convert data into a form that a computer will recognise and can process. • Use binary code to create an on/off picture. Refer to Using binary to create on/off pictures (http://www.digitaltechnologieshub.edu.au/teachers/lesson-ideas/using-binary-to-create-on-off-pictures). • Converting numbers and text to and from binary/ASCII code using given tables: ▪ Complete Lesson 1 Binary Baubles – convert text to binary using ASCII Encoder cards. Refer to Binary Baubles (https://aca.edu.au/resources/baubles/). ▪ Students use the ASCII Encoder Cards from the previous letters provided to write a message in code, and then share with others to decode the message. • Students create an educational video explaining binary or data representation. This should be created in stop-motion style or with still images, rather than video/film format. This will allow students to understand that the use of still images can create a story or represent a theme. Music or voice-overs could also be considered. ▪ This task is designed to be a collaborative project. ▪ Software can include Adobe Premiere®, Windows® Movie Maker or simply PowerPoint (Auto transitions with quick time transitions between slides).

Weeks	Syllabus content	Content unpacked	Suggested teaching and learning experiences
			Sample storyboard for film and animation 
9–10	**Collecting, managing and analysing data** Collect, sort, interpret and visually present different types of data using software to manipulate data for a range of purposes	• Data gathering can be from primary or secondary data sources. Primary data is collected by the students themselves. Secondary data is collected from third party sources and is used to analyse a situation or problem. • Interpretation of data refers to the reviewing of collected data to view patterns and trends and, with this knowledge, make assumptions about the data or future data. • The manipulation of data can be the changing of data based on parameters given or retrieving the data that is required.	• Students are given two sets of unorganised and uncategorised data. They are required to sort the data into numeric or alphabetical order. Students then create graphs of their choice on this data. They analyse the data and present their findings to the class. Students will have chosen different graphs and data that interests them. Undertake a class discussion about how to interpret data differently. • Students view the Photo Story 3 tutorial (https://www.youtube.com/watch?v=_u3HDMfXpdI) to help them complete the task. • Additional activity: School Curriculum and Standards Authority – Graphs from the Judging Standards section of the extranet (access requires registration).
11–12	**Digital implementation** Design, modify, follow and represent both diagrammatically, and in written text, simple algorithms (sequence of steps) involving branching (decisions) and iteration (repetition) Manage the creation and communication of information, including online collaborative projects, using agreed social, ethical and technical protocols **Investigating and defining** Define a problem, and a set of sequenced steps, with users making decisions to create a solution for a given task	• Sequence of steps refers to the order of a program. This must be logical to avoid any logic errors. • Branching is a term that refers to having more than one intended outcome. • The digital solution produced must include decisions made by the user, which are referred to as selection. The three types of selection are: ▪ one-way selection (if-then) ▪ two-way selection (if-then-else) ▪ multiway selection (CASE). • Iteration is the inclusion of repetition or loops. • Iteration includes: ▪ test first (do while) ▪ test last (repeat loop) ▪ fixed (for loop) • Simple visual programing environments include simple block based programming, such as Scratch™. This allows for students to learn the skills of computational thinking without constraints of syntax rules, such as spelling. • Inputs can be received in a variety of ways, including mouse, keyboard, speech and touchscreen.	• Individual activity – Journaling toward algorithmic thinking (https://docs.google.com/document/d/1Qouj-ZxcPVmYehvIvLGnNV0X_4E_9YNyjXEeCOmmBaI/edit#heading=h.trqxxjck5x44). • Lego™ (or block) model activity [may be completed using actual blocks or Lego Digital Design software (https://www.lego.com/en-us/ldd/download)]. • Students write their own series of instructions to create a simple Lego (or block) model. • Swap instructions with another student. • Without asking questions, students are to create the model from the given instructions. • Compare the model created with the original model – did the instructions provide all the necessary detail to create the model accurately? • Useful links: ▪ Lucidchart – Sample Flowchart Template (https://www.lucidchart.com/pages/templates/flowchart/sample-flowchart-template) ▪ Lucidchart – Computer Flowchart Template (https://www.lucidchart.com/pages/templates/flowchart/computer-flowchart-template) ▪ BBC Northern Ireland – Sample Storyboard (http://www.bbc.co.uk/northernireland/myplacemyspace/promote-your-day-out/with-film/sample-storyboard.shtml). ▪ Storyboard examples for children (http://tarese.alianzasistema.org/storyboard-examples-for-children/)

Weeks	Syllabus content	Content unpacked	Suggested teaching and learning experiences

	Identify available resources
	
	Producing and implementing
	Select, and apply, safe procedures when using a variety of components and equipment to make solutions
		• Technical protocols refer to students' agreed set of rules when using online collaboration. These can simply include respecting hardware, the school's information technology (IT) code of conduct or netiquette.
		• Management of this information should include the validity and referencing of any data gathered.
		• Common symbols in a flowchart and what they represent; for example:
		▪ start and end (rounded rectangle or oval)
		▪ process (rectangle)
		▪ decision (diamond)
		▪ input/output (parallelogram)
		▪ line connectors (arrow).
		• Common content included in a storyboard, such as:
		▪ sketches of key scenes/slides
		▪ main event in each scene or content of slides
		▪ actions that will take place
		▪ audio and/or text to be added.
		• The sequence developed must be in a logical order. Computers work sequentially, line by line, top to bottom. If the order is incorrect, a logic error will be given. Students cannot put shoes on before socks, a computer works in the same way. Students need to understand that logical sequence is very important in programming.
		• Solutions need to be innovative and incorporate problem-solving skills.
		• Available resources should include hardware and software. Students need to justify why they have chosen selected hardware or software.
			• Given the flowchart provided, discuss whether students think that they would be able to follow the plan.
			• Cut out large shapes on cardboard and add content to the shapes when relevant.
			• Students create a flowchart and/or storyboard to demonstrate a process or design an animation. Refer to Alice – Tutorial: Designing and Animation (https://www.alice.org/resources/exercise-and-project/tutorial-designing-an-animation/).
			• Based on the previously created non-linear PowerPoint Quiz for Digital Systems, students are to create a flowchart to explain the decisions (branching) and the logical sequence of their quiz. This can be completed individually under test conditions to validate prior knowledge.
			
			Example of a simple two way selection
			 
			Two-way selection
			If *condition* then .. else
			
			Input (Age)
			If (Age >= 16) and (Age <= 65) then
			Price ← 35
			Else
			Price ← 20
			End If
			Output ('The cost will be $', Price)

13–15	**Digital implementation** Implement and use simple visual programming environments that include branching (decisions), iteration (repetition) and user input	• A visual programming language is one that allows the user to create programmes or code using graphic elements rather than text; for example: ▪ Scratch ▪ Kodu™ ▪ Alice™ ▪ Blocky Games: Turtle™. • Begin to introduce subject and software specific terminology as appropriate to selected software and task; for example: ▪ programming ▪ code ▪ loops/iteration ▪ branching/decisions.	• Introduce selected visual programming tool/software and explore the user interface. • Practise simple algorithms to create a visual effect; for example, draw a simple shape, move an object around the screen. • Students complete selected tutorials to develop their programming skills in the selected software; for example: ▪ Scratch (https://scratch.mit.edu/help/videos/) ▪ Kodu (https://www.kodugamelab.com/) ▪ Alice (http://www.alice.org/resources/alice-3-how-tos/) ▪ Blocky Games Turtle (https://blockly-games.appspot.com/turtle?lang=en&level=1). • Activity for planning and creating basic program idea: ▪ School Curriculum and Standards Authority – Scratch it! from the Judging Standards section of the extranet (access requires registration). • Other ideas for digital implementation include: ▪ creating a game to help other students increase literacy and numeracy ▪ creating a game to promote sustainability or recycling ▪ creating a game to promote different cultures from around the world.

Weeks	Syllabus content	Content unpacked	Suggested teaching and learning experiences
16	**Evaluating** Develop collaborative criteria to evaluate and justify design processes and solutions **Collaborating and managing** Work independently, or collaboratively when required, considering resources, to plan, develop and communicate ideas and information for solutions	• Evaluation can be formative or summative or both. • Developing a criteria for evaluation should be conducted as a class or individually. • Peer evaluation of designs is very beneficial to students. • Sample criteria used to evaluate the process may include: ▪ Did I work well together and communicate with my partner? ▪ Did I stay on task? ▪ Did I meet the task requirements? ▪ What issues did I encounter? ▪ How did I overcome the issues? ▪ Did I complete the task in the time given? • Criteria used to evaluate solutions (i.e. the quiz) may include: ▪ suitability and/or clarity of questions ▪ variety of questions ▪ working hyperlinks ▪ suitability of images ▪ accuracy of feedback (e.g. Was a wrong answer marked as correct?)	• Create a class blog: Class blog (https://www.digitaltechnologieshub.edu.au/teachers/lesson-ideas/class-blog). • Students complete the final part of the task – evaluations. • As a class, develop criteria to evaluate the processes used to complete the task. ▪ Each student completes a self-evaluation of the processes they implemented to complete the task based on the criteria developed. • As a class, develop criteria to evaluate the solution to the task, i.e. the quiz produced. ▪ Each student completes a self-evaluation of their own quiz based on the criteria developed. • Students attempt quizzes developed by other students (accessed online via the collaborative site determined by the teacher) and evaluate them based on the criteria developed.
17–18	**Digital implementation** Manage the creation and communication of information, including online collaborative projects, using agreed social, ethical and technical protocols	• Copyright status of online images. • Check usage rights of image search. • Suitable sites for images without copyright. • Breaches of copyright law. • Plagiarism is one of the most common forms of copyright breaches for students. Plagiarism refers to using someone else's ideas or work without asking or acknowledging where it came from.	• Locate suitable images of computer components to add to the online collaborative task from the previous lesson. • Find an example of a breach of copyright, and analyse and discuss the outcome. • Useful websites for teaching about copyright include: ▪ Teaching Treasures (https://www.teachingtreasures.com.au/Danger%20zone/copyright-activity.htm) ▪ The Game Is On! (https://www.copyrightuser.org/educate/the-game-is-on/) ▪ Crazy4computers.net (http://www.crazy4computers.net/copyright-law--fair-use.html). • Students create a short (30-second) song/sound bite that efficiently warns against the consequences of breaching the *Copyright Act 1968* or plagiarism. This can be created individually or collaboratively. A plan or script should be created prior to commencement and all sounds or music snippets used need to be correctly referenced. Software to aid students in the development of this task can include GarageBand®, Soundation, Audacity™, Acid Xpress or Myna.

Note: the above Teaching and Learning Outline is designed for two hours per week for 18 weeks for a total of 36 hours.