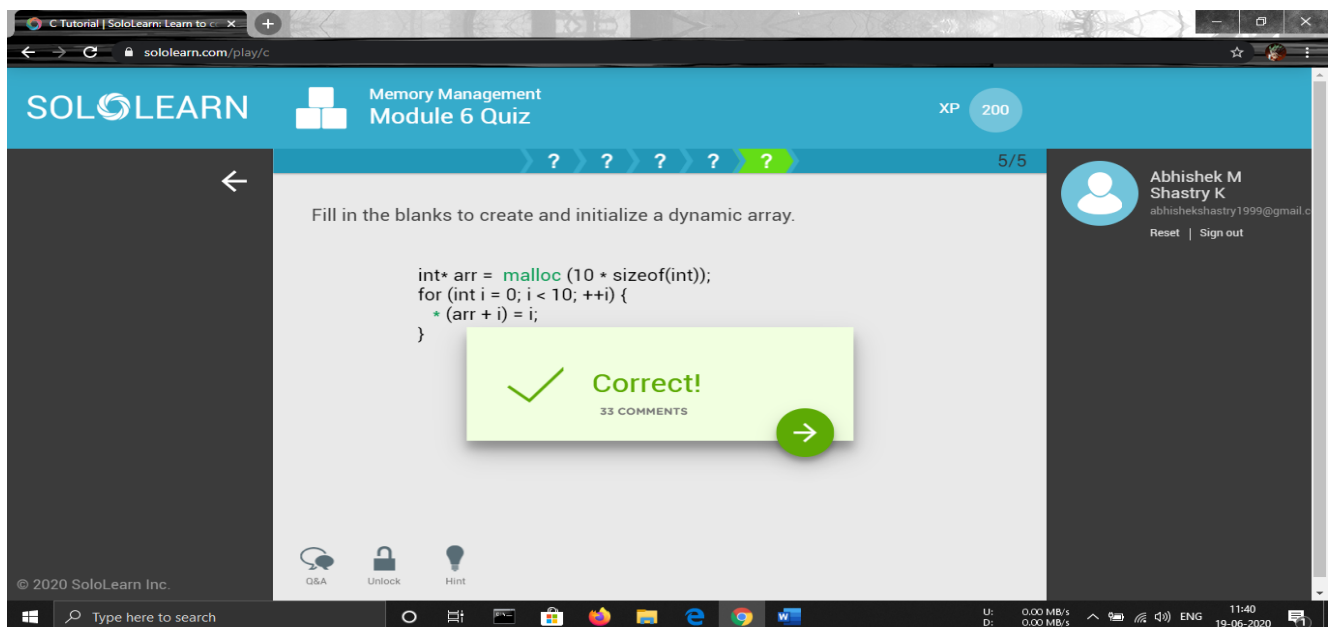
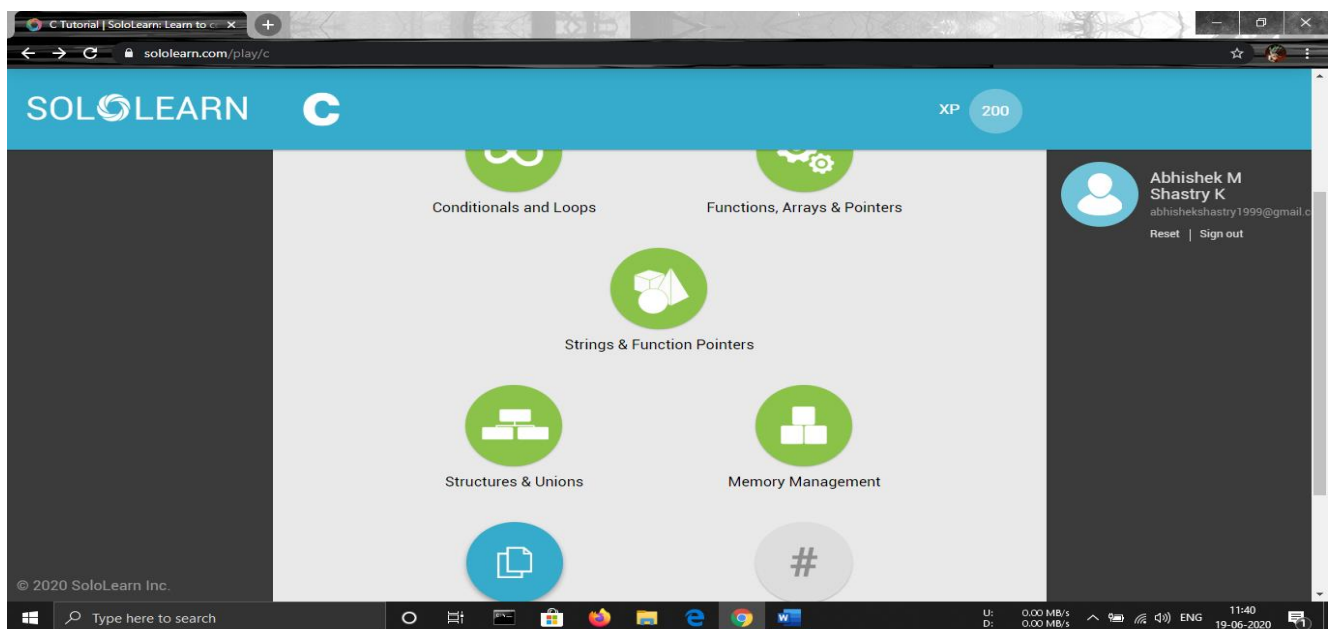


DAILY ASSESSMENT REPORT

Date:	19/06/2020	Name:	Abhishek M Shastry K
Course:	C Tutorial by SOLOLEARN	USN:	4AL17EC002
Topic:	1] Structures & Unions 2] Memory Management	Semester & Section:	6 th 'A'
Github Repository:	AbhishekShastry-Courses		

FORENOON SESSION DETAILS

Image of session



Report

Structures & Unions

- A **structure** is a user-defined data type that groups related variables of different data types.
- A **structure** declaration includes the keyword `struct`, a structure tag for referencing the structure, and curly braces `{ }` with a list of variable declarations called members.
- To declare variables of a structure data type, use the keyword `struct` followed by the struct tag, and then the variable name. You access the members of a struct variable by using the **. (dot operator)** between the variable name and the member name.
- The **typedef** keyword creates a type definition that simplifies code and makes a program easier to read. `typedef` is commonly used with structures because it eliminates the need to use the keyword `struct` when declaring variables.
- A function can have structure parameters that accept arguments by value when a copy of the structure variable is all that is needed. For a function to change the actual values in a struct variable, pointer parameters are required.
- An array can store elements of any data type, including structures. After declaring an array of structures, an element is accessible with the index number.
- A **union** allows to store different data types in the same memory location. It is like a structure because it has members. However, a union variable uses the same memory location for all its member's and only one member at a time can occupy the memory location.
- A **union** declaration uses the keyword `union`, a union tag, and curly braces `{ }` with a list of members. Union members can be of any data type, including basic types, strings, arrays, pointers, and structures. You access the members of a union variable by using the **. dot operator** between the variable name and the member name.
- When assignment is performed, the union memory location will be used for that member until another member assignment is performed.
- A **pointer to a union** points to the memory location allocated to the union.
- A **union pointer** is declared by using the keyword `union` and the union tag along with `*` and the pointer name.
- An array can store elements of any data type, including unions. With unions, it is important to keep in mind that only one member of the union can store data for each array element.

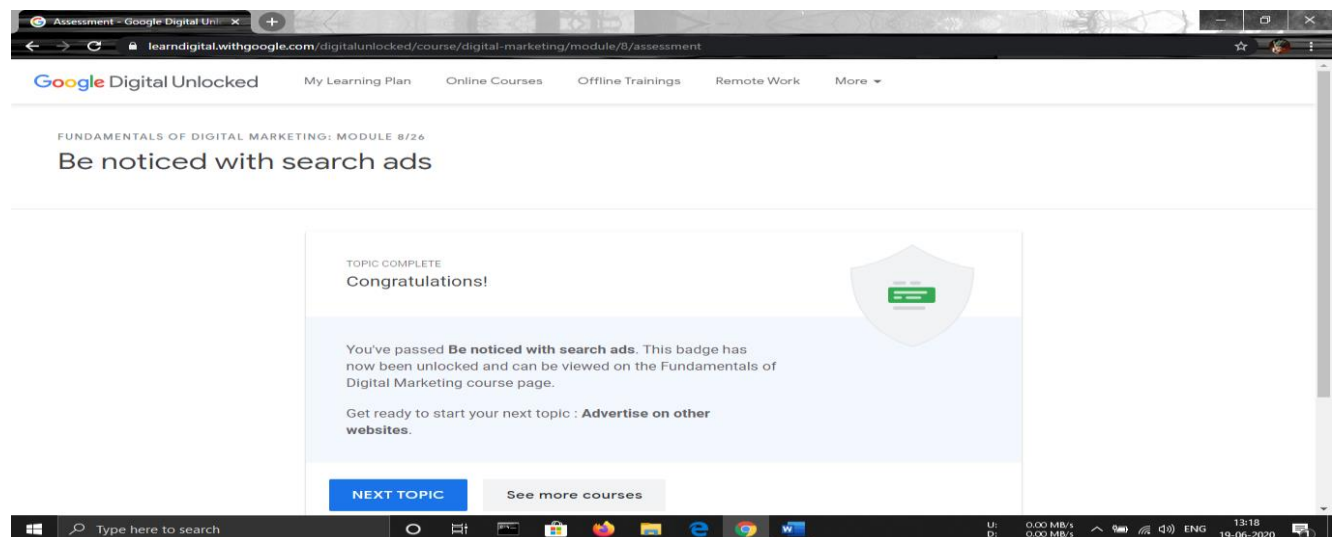
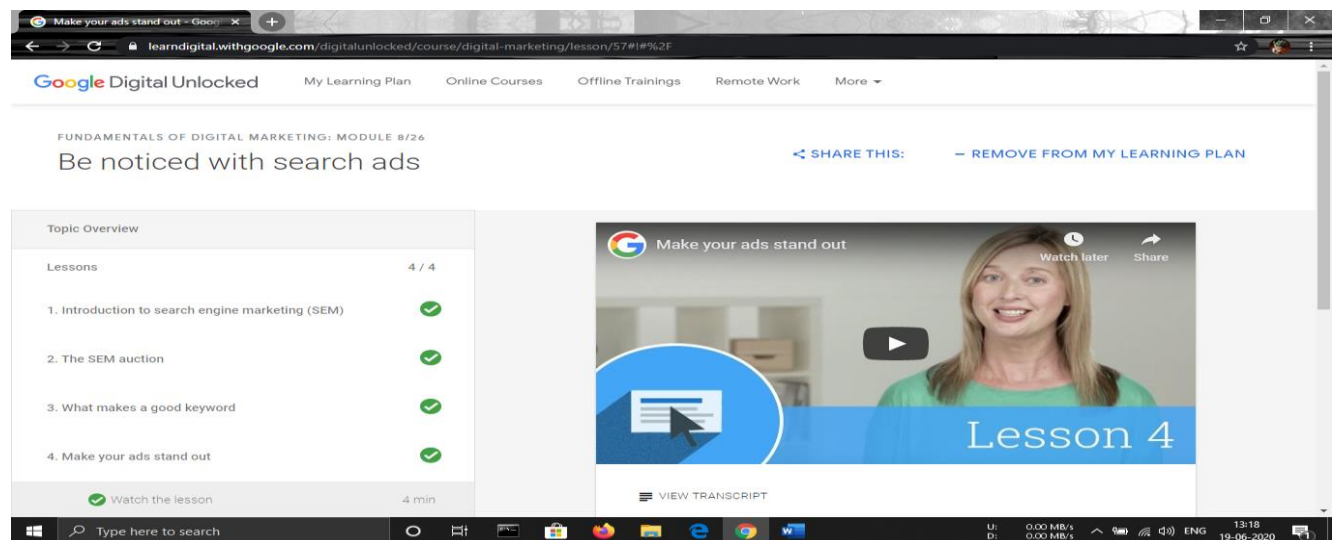
Memory Management

- Understanding **memory** is an important aspect of C programming. When you declare a variable using a basic data type, C automatically allocates space for the variable in an area of memory called the **stack**. An int variable, for example, is typically allocated 4 bytes when declared. We know this by using the **sizeof** operator.
- The **stdlib.h** library includes memory management functions.
- The statement **#include <stdlib.h>** at the top of your program gives you access to the following:
 - ✓ **malloc(bytes)**: Returns a pointer to a contiguous block of memory that is of size bytes.
 - ✓ **calloc(num_items, item_size)**: Returns a pointer to a contiguous block of memory that has num_items items, each of size item_size bytes. Typically used for arrays, structures, and other derived data types. The allocated memory is initialized to 0.
 - ✓ **realloc(ptr, bytes)**: Resizes the memory pointed to by ptr to size bytes. The newly allocated memory is not initialized.
 - ✓ **free(ptr)**: Releases the block of memory pointed to by ptr.
- The **allocated memory** is contiguous and can be treated as an array. Instead of using brackets [] to refer to elements, pointer arithmetic is used to traverse the array. You are advised to use + to refer to array elements. Using ++ or += changes the address stored by the pointer.
- If the allocation is unsuccessful, NULL is returned. Because of this, you should include code to check for a NULL pointer.
- When allocating memory for a **string pointer**, you may want to use string length rather than the sizeof operator for calculating bytes.
- Many algorithms implement a **dynamic** array because this allows the number of elements to grow as needed. Because elements are not allocated all at once, dynamic arrays typically use a structure to keep track of current array size, current capacity, and a pointer to the elements.

Date:	19/06/2020	Name:	Abhishek M Shastry K
Course:	Google Digital Unlocked: Fundamentals of digital marketing	USN:	4AL17EC002
Topic:	1] Be noticed with search ads <ul style="list-style-type: none"> • Introduction to search engine marketing (SEM) • The SEM auctions • What makes a good keyword? • Make your ads stand out 	Semester & Section:	6th 'A'
Github Repository:	AbhishekShastry-Courses		

AFTERNOON SESSION DETAILS

Image of session



Report

Introduction to search engine marketing (SEM)

- That's the big difference when comparing SEM to traditional forms of advertising, like ads in newspapers and magazines, or billboard posters. If you advertise wedding photography using those, the ad is shown to a bunch of people whether or not they want wedding photos. And you pay no matter what.
- With SEM, it's very likely that people who clicked your ad are interested in what you have to offer, because they told you so. Or rather, they told the search engine when they searched for wedding photographer Cardiff, saw your ad and clicked on it. Unlike traditional marketing, SEM targets people actively looking for products and services. This is different from broadcasting a message to people that they may or may not have any interest in.
- Search engine marketing is a uniquely powerful option for your business. It's simple, but effective: People search for things they want. And advertisers target specific searches, but only pay if someone clicks to learn more. That's search engine marketing.

The SEM auctions

- Search engines limit the number of slots adverts can appear in on an individual page. Advertisers then compete in an auction for those slots.
- When two competing businesses have equal bid prices, the business with a higher Quality Score will appear higher on the search results page. One of the best ways you can improve your performance in SEM is to consistently review your campaigns to find ways to become more relevant.
- This is to ensure the most relevant adverts win the auction, not simply the ones with the highest bids. If your ads lack relevance and get a low-Quality Score, it'll be difficult to compete. In fact, if relevance is too low, the search engines may not display your ads at all.
- To improve your performance, stay focused on improving the relevance of all your adverts. Make sure that your adverts always closely match the terms that people have just searched for. By doing this, you can win better positions for your ads, without having to pay more money.

What makes a good keyword?

- Keywords that indicate a person looking to hire a wedding photographer in Cardiff are clearly relevant. However, keywords that suggest a person looking to do their own wedding photography, or that suggest digital photography when you only shoot in film, are far less relevant.
- Try to focus solely on keywords that are relevant to your business, and you'll have more success going forward. Another factor to consider is how much traffic a given keyword is likely to get.
- This might describe your business exactly, but with so much specific detail it's unlikely that many people will search for it. Bidding on this keyword probably won't result in many people finding your business.
- Keyword research tools can provide traffic estimates to help you get a handle on this. But in reality, you'll probably have to use a bit of trial and error to figure out which keywords are relevant and get sufficient traffic.
- It's a bit of a balancing act, but weighing these three factors can set your ad campaign off to the right start. Be sure to monitor your keywords closely - if you do, you'll quickly learn about what's working, and then you can make adjustments as you move forward.

Make your ads stand out

- Relevance is the name of the game in search engine marketing. Try and write your advert's headline to match the searcher's words as closely as possible.
- If you're running a sale, or have a special benefit the customer can take advantage of, that's a really easy way to attract attention. Let's take your first description line, which said Photos for All Weddings, and update it with a special offer. How about this: Get 25% Off Your Photo Collage.
- Using a few simple guidelines can help your adverts stand out from the competition. First, you want to customize your advert headline to the search. Then include a special or timely offer. And finally, give a compelling call to action.
- These techniques can help improve your ads across the board. Just remember that writing great adverts is an ongoing process - just like researching keywords or monitoring your bid prices. Focus on these tips over the long run, and you're likely to see success.