Solutions

I MID TERM Examination

II SEM

Subject: Computer Programming II

Section - A

Answer 1:

 (a)

System software is a type of computer [program](http://searchsoftwarequality.techtarget.com/definition/program) that is designed to run a computer’s hardware and [application programs](http://searchsoftwarequality.techtarget.com/definition/application-program). If we think of the computer system as a layered model, the system [software](http://searchsoa.techtarget.com/definition/software) is the interface between the hardware and user applications.

The operating system ([OS](http://searchcio-midmarket.techtarget.com/definition/operating-system)) is the best-known example of system software.

 (b)

A compiler is [computer software](https://en.wikipedia.org/wiki/Computer_software) that transforms computer code written in one [programming language](https://en.wikipedia.org/wiki/Programming_language) (the source language) into another programming language (the target language). Compilers are a type of [translator](https://en.wikipedia.org/wiki/Translator_%28computing%29) that support digital devices, primarily computers. The name *compiler* is primarily used for programs that translate [source code](https://en.wikipedia.org/wiki/Source_code) from a [high-level programming language](https://en.wikipedia.org/wiki/High-level_programming_language) to a [lower level language](https://en.wikipedia.org/wiki/Lower_level_language) (e.g., [assembly language](https://en.wikipedia.org/wiki/Assembly_language), [object code](https://en.wikipedia.org/wiki/Object_code), or [machine code](https://en.wikipedia.org/wiki/Machine_code)) to create an [executable](https://en.wikipedia.org/wiki/Executable) program.

(c)

we are directly assigning String to variable by writing text in double quotes.

In this type of initialization , we don’t need to put NULL or Ending / Terminating character at the end of string. It is appended automatically by the compiler.

char name [ ] = "RIET";

 (d)

Pointer declaration : the syntax to declare a pointer

 *data\_type* \**poiter\_name*;

Let's consider with following example statement

int \*ptr;

Here, in this statement

* ptr is the name of pointer variable (name of the memory blocks in which address of another variable is going to be stored).
* The character asterisk (\*) tells to the compiler that the identifier ptr should be declare as pointer.
* The data type int tells to the compiler that pointer ptr will store memory address of integer type variable.

(e )

The commonly used control strings are :

|  |  |  |
| --- | --- | --- |
| Specifier | Description | Example |
| f | Display the floating point number using decimal representation | 3.1415 |
| e | Display the floating point number using scientific notation with e | 1.86e6 (same as 1,860,000) |
| E | Like e, but with a capital E in the output | 1.86E6 |
| g | Use shorter of the two representations: f or e | 3.1 or 1.86e6 |
| G | Like g, except uses the shorter of f or E | 3.1 or 1.86E6 |

c character a

d Decimal integer 782

s string RIET

(f)

To perform bit-level operations in C programming, bitwise operators are used.

| Operators | Meaning of operators |
| --- | --- |
| & | [Bitwise AND](https://www.programiz.com/c-programming/bitwise-operators#and) |
| | | [Bitwise OR](https://www.programiz.com/c-programming/bitwise-operators#or) |
| ^ | [Bitwise XOR](https://www.programiz.com/c-programming/bitwise-operators#xor) |
| ~ | [Bitwise complement](https://www.programiz.com/c-programming/bitwise-operators#complement) |
| << | [Shift left](https://www.programiz.com/c-programming/bitwise-operators#left-shift) |
| >> | [Shift right](https://www.programiz.com/c-programming/bitwise-operators#right-shift) |

(g)

Function call by value is the default way of calling a function in C programming.

In this, The value of actual parameter assigns to formal parameter.

Actual parameters: The parameters that appear in function calls.
Formal parameters: The parameters that appear in function defination.

For example:

#include <stdio.h>

int sum(int a, int b)

{

 int c=a+b;

 return c;

}

int main(

{

 int var1 =10;

 int var2 = 20;

 int var3 = sum(var1, var2);

 printf("%d", var3);

 return 0;

}

In the above example variable a and b are the formal parameters (or formal arguments). Variable var1 and var2 are the actual arguments (or actual parameters). The actual parameters can also be the values. Like sum(10, 20), here 10 and 20 are actual parameters.

(h)

 getchar(): single character input function.

Syntax:

int getchar(void);

Example:

|  |
| --- |
| // Example for getchar() in C#include <stdio.h>int main(){   printf("%c", getchar());   return 0;} |

Run on IDE

Input: g(press enter key)

Output: g

Section B

Answer 2 (a)

Arrays a kind of data structure that can store a fixed-size sequential collection of elements of the same type. An array is used to store a collection of data, but it is often more useful to think of an array as a collection of variables of the same type.

Instead of declaring individual variables, such as number0, number1, ..., and number99, we declare one array variable such as numbers and use numbers[0], numbers[1], and ..., numbers[99] to represent individual variables. A specific element in an array is accessed by an index.

All arrays consist of contiguous memory locations. The lowest address corresponds to the first element and the highest address to the last element.

The name of array represents address.

To access individual item of 1- D array by address,

\*(array name+i);

Where, i= 0,1,2……(size-1).

Answer 2 (b)

**Linear search C program**

#include <stdio.h>

int main()

{

 int array[100], search, c, n;

 printf("Enter the number of elements in array**\n**");

 scanf("%d",&n);

 printf("Enter %d integer(s)**\n**", n);

 for (c = 0; c < n; c++)

 scanf("%d", &array[c]);

 printf("Enter the number to search**\n**");

 scanf("%d", &search);

 for (c = 0; c < n; c++)

 {

 if (array[c] == search) */\* if required element found \*/*

 {

 printf("%d is present at location %d.**\n**", search, c+1);

 **break**;

 }

 }

 if (c == n)

 printf("%d is not present in array.**\n**", search);

 return 0;

}

Answer 3 (a)

The function takes a single argument, i.e, the string variable whose length is to be found, and returns the length of the string passed.

The strlen() function is defined in [<string.h>](https://www.programiz.com/c-programming/library-function/string.h) header file.

### Example: C strlen() function

#include <stdio.h>

#include <string.h>

int main()

{

 char a[20]="Program";

 char b[20]={'P','r','o','g','r','a','m','\0'};

 char c[20];

 printf("Enter string: ");

 gets(c);

 printf("Length of string a = %d \n",strlen(a));

 //calculates the length of string before null charcter.

 printf("Length of string b = %d \n",strlen(b));

 printf("Length of string c = %d \n",strlen(c));

 return 0;

}

**Output**

Enter string: String

Length of string a = 7

Length of string b = 7

Length of string c = 6

Answer 3 (b)

The C library function **char \*strcpy(char \*dest, const char \*src)** copies the string pointed to, by **src** to **dest**.

## Declaration

Following is the declaration for strcpy() function.

char \*strcpy(char \*dest, const char \*src)

## Parameters

* **dest** − This is the pointer to the destination array where the content is to be copied.
* **src** − This is the string to be copied.

## Return Value

This returns a pointer to the destination string dest.

The following example shows the usage of strcpy() function.

#include <stdio.h>

#include <string.h>

int main () {

 char src[40];

 char dest[100];

 strcpy(src, "This is my college riet");

 strcpy(dest, src);

 printf("Final copied string : %s\n", dest);

 return(0);

}

Answer 4

The output of given program:

5 10

Answer 5

The output of given program:

 4