

CSE S/A

2019

1a) EEPROM - Electrically alterable read-only memory.

PROM - Programmable read-only memory.

DLT - Digital Linear Type

SRAM - Static random access memory

1b) OS:

An operating system is a software programme require to manage and operate a computing device like smartphone, tablets, computers, supercomputers, web servers, cars, network towers, smartwatches etc. It's the operating system that eliminates the need to know coding language to interact with computing devices. It's a layer of graphical user interface (GUI) which acts as a platform between the user and the computer hardware. Moreover the operating system manages the software side of a computer and controls program execution.

Linux:

Linux is an open-source operating system (OS). ~~one operating~~
~~systems~~ It's an entire family of open source Unix
operating systems, that are based on the Linux kernel.

Windows:

Windows is a graphical operating system developed by Microsoft. It allows users to view and store files, run the software; play games, watch videos, and provide a way to connect to the internet. It was released for both home computing and professional works.

DOS:

DOS - Disk Operating System, was the first operating system used by IBM-compatible computers. It was originally available in two versions that were essentially the same, but marketed under two different names: PC-DOS & MS-DOS.

2019_1.cpp # X

2019_1_c

(Global Scope)

main()

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5
6     double F, C;
7     cout << "Enter the value in Fahrenheit degree: ";
8     cin >> F;
9     C = (5.0 / 9.0) * (F - 32.0); //use decimal point
10    cout << "The value in Celsius is: " << C << endl;
11
12    return 0;
13 }
```

Ques) Translation Program

A translator is a programming language processor that converts a computer program from one language to another : It takes a program written in source code and convert it into machine code. It ~~also~~ discovers and identifies the errors during translation.

It translate high level language program into a machine language program that the central processing unit (CPU) can understand and detect errors.

Compiler

Compiler scans the whole program in one go.

As it scans the code in one go the errors (if any) are shown at the end together.

Interpreter

Translates program one statement at a time.

Considering it scans code one line at a time , errors are shown line by line.

[geeks for geeks]

Continue .

```
2019_2_b.cpp # x
2019_2_b (Global Scope) main()

1 #include <iostream>
2 using namespace std;
3 int main() {
4
5     int mat1[3][3] = { {0} }, mat2[3][3] = { {0} }, mat3[3][3] = { {0} }, i = 0, j = 0, k = 0;
6
7
8     cout << "\nEnter values for first 3 x 3 matrix:\n";
9     for (i = 0; i < 3; i++){
10         for (j = 0; j < 3; j++)
11             cin >> mat1[i][j];
12     }
13
14     cout << "\nEnter values for second 3 x 3 matrix:\n";
15     for (i = 0; i < 3; i++){
16         for (j = 0; j < 3; j++)
17             cin >> mat2[i][j];
18
19     for (int i = 0; i < 3; i++)
20         for (int j = 0; j < 3; j++) {
21             for (int k = 0; k < 3; k++)
22                 mat3[i][j] += mat1[i][k] * mat2[k][j];
23     }
```

2019_2_b.cpp # X

2019_2_b (Global Scope) main()

```
22     } mat3[i][j] += mat1[i][k] * mat2[k][j];
23 }
24
25 cout << "\nThe product of the above two matrices is:\n";
26 for (i = 0; i <= 2; i++){
27     for (j = 0; j <= 2; j++) {
28         cout << "\t" << mat3[i][j];
29     }
30     cout << "\n";
31 }
32
33 return 0;
34 }
```

20)

Software:

Software is a set of instruction, data or programs used to operate computer and execute specific tasks. It is the opposite of hardware, which describe the physical aspect of a computer. More on internet...

There are two types of software,

① Application Software

② System Software

System Soft.

They are designed to manage the resources of the system

like memory and process management

Application soft

They are designed to fulfill the requirements of the user for performing specific tasks.

gurug.com blog

continue

3(a) The main difference between `break` and `continue` is that `break` is used for immediate termination of loop. On the other hand '`continue`' terminate the current iteration and resume the control of the next iteration of the loop.

More on- tech-difference.com

4(a) The process in which a function calls itself directly or indirectly is called recursion and the corresponding function is called recursive function : Using recursive algorithm certain

```
2019_3_b.cpp ✘ X
2019_3_b (Global Scope) main()
```

```
4 int main()
5 {
6     int i, j, rows;
7     cout << "Enter number of rows : " << endl;
8     cin >> rows; //number of rows
9
10    for (i = 1; i <= rows; i++){
11        for (j = i; j < rows; j++){ /* Print trailing spaces */
12            cout << " ";
13        }
14        for (j = 1; j <= (2 * i - 1); j++){ /* Print hollow pyramid */
15            /*
16             --Print "2" for last row (i==rows),
17             --first column(j==1) and
18             --for last column (j==(2*i-1)).
19             */
20            if (i == rows || j == 1 || j == (2 * i - 1)){
21                cout << "2";
22            }
23            else{
24                cout << " ";
25            }
26        }
27        cout << endl; /* Move to next line */
28    }
29
30    return 0;
31 }
```

```
1  #include <iostream>
2  using namespace std;
3
4  int main()
5  {
6      int i, j, rows;
7      cout << "Enter number of rows : " << endl;
8      cin >> rows; //number of rows
9
10     for (i = 1; i <= rows; i++){
11         for (j = i; j < rows; j++){ /* Print trailing spaces */
12             cout << " ";
13         }
14         for (j = 1; j <= (2 * i - 1); j++){ /* Print hollow pyramid */
15             /*
16              --Print "2" for last row (i==rows),
17              --first column(j==1) and
18              --for last column (j==(2*i-1)).
19             */
20             if (i == rows || j == 1 || j == (2 * i - 1)){
21                 cout << "2";
22             }
23             else{
24                 cout << " ";
25             }
26         }
27         cout << endl; /* Move to next line */
28     }
29
30     return 0;
31 }
```

p1.cpp X

2019_3_c_p1 (Global Scope) main()

```
1 #include <iostream>
2 using namespace std;
3 int main() {
4
5     int a;
6     for (a = 2; a < 88; a++) {
7         if ((a % 2 == 0) || (a % 5 == 0)) {
8             continue; //breaks the iteration and skip that number
9         }
10        cout << a << endl;
11    }
12
13    return 0;
14 }
```

Microsoft Visual Studio Debug Console

9
11
13
17
19
21
23
27
29
31
33
37
39
41
43
47
49
51
53
57
59
61
63
67
69
71
73
77
79
81
83
87

133 % No issues found

Output

Show output from: Debug

The thread 0x35ec has exited with code 0 (0x0).
The program '[11564] 2019_3_c_p1.exe' has exited with code 0 (0x0).

Scanned with CamScanner

p2.cpp X

2019_3_c_p2 (Global Scope) main()

```
1 #include <iostream>
2 using namespace std;
3 int main() {
4     int x;
5     for (x = 1; x < 33; x++) {
6         if (x % 3 == 1) {
7             break; //breaks the whole loop
8         }
9         cout << x << endl;
10    }
11
12
13    return 0;
14 }
```

Microsoft Visual Studio Debug Console

E:\kaisarTMP\CSE Exam Code\2019_3_c_p2\x64\Debug\2019_3_c_p2.exe
To automatically close the console when debugging stops, enable
le when debugging stops.
Press any key to close this window . . .

133 % No issues found

Output

iteration and resume the execution at the next iteration of the loop.

More on - tech-difference.com

- Q) The process in which a function calls itself directly or indirectly is called recursion and the corresponding function is called recursive function : Using recursive algorithm certain problem can be solved quite ~~easy~~ easily.

Advantages.

- ① The code may be easier to write
- ② To solve such problems which are naturally recursive such as tower of hanor
- ③ Reduce unnecessary calling of function.
- ④ Extremely useful when applying the same soln.
- ⑤ Recursion reduce the length of code.

- ⑥ Really useful for solving the data structure problem.
- ⑦ Stacks operations and infix, prefix, postfix evaluation etc.
- ⑧ Recursion adds more clarity to the code and easier to debug.
- ⑨ Reduce the time complexity.
- ⑩ Perform better in solving tree structure program.

5a) A number system can be considered as a mathematical notation of numbers using a set of digits or symbols.

A 3 base number system means the base value of a number system is 3, then it's known as a ternary representation. The digits in the system are 0, 1, 2.

```
1 #include <iostream>
2 using namespace std;
3 int main() {
4
5     double number;
6     int count; int count;
7     do {
8         cout << "Enter a Number= ";
9         cin >> number;
10    } while (count < 6);
11
12    return 0;
13 }
```

```
1 #include <iostream>
2 using namespace std;
3 int main() {
4
5     double number;
6     int count = 0; //value initialized
7     do {
8         cout << "Enter a Number= ";
9         cin >> number;
10        count++; //value increment
11    } while (count < 6);
12
13    return 0;
14 }
```

```
1 #include <iostream>
2 using namespace std;
3
4 int fac(int n) {
5     if (n > 1) {
6         return n * fac(n - 1);
7     }
8     else {
9         return 1;
10    }
11 }
12 int main() {
13
14     int i, n;
15     double sum = 0;
16     cout << "Enter the value of n: ";
17     cin >> n;
18     for (i = 0; i <= n; i++) {
19         sum += (1.0 / fac(i)); //series of e
20     }
21
22     cout << sum;
23
24     return 0;
25 }
```

etc.

- ⑧ Recursion adds more clarity to the code and easier to ~~and~~ debug.
- ⑨ Reduce the time complexity.
- ⑩ Perform better in solving tree structure program

- 5a) A number system can be considered as a mathematical notation of numbers using a set of digits or symbols.
- A 3 base number system means the base value of a number system is 3, and it's known as a ternary representation. The digits in the system are 0, 1, 2.
- A 5 base number system means the base value of a number system is 5, and it's known as quinary representation. The digits in the system are 0, 1, 2, 3, 4.

6a) We need both public and private members in a class because by default all the members of a class are private and the methods are declared as public ~~as~~ in class as the method will be called later.

P_uM

It's an access modifier which specifies the access restriction to class members.

The public members are accessible from anywhere outside of the class.

P_rM

The private member variable or function is not accessed or viewed from outside the class.

discontinued.

6b) Destructor is an instance member function which is invoked automatically whenever an object is going to be destroyed. Meaning, a destructor is the last function that is going to be called before an object is destroyed.

Properties

- * Destructor function is automatically invoked when the object are destroyed.
- * It can't be declared static or const.
- * The destructor does not have arguments.
- * It has no return type not even void.
- * An object of a class with a destructor can't become a member of the union.
- * A destructor should be declared in the public section of the class.
- * The programmers can't access the address of destructor.
- * The main ~~purpos~~ purpose of the destructor of a class is to free computer memory.

+a) 1-D Array

It is considered as the "list of variables of ~~similar~~ similar data types"; and each variable can be distinctly accessed by specifying its index in square brackets preceded by the name of the array.

Example

~~top~~ Data-type variable-name [size];

int month[12]

then ex[10] = { 'J', 'F', 'M', 'A', 'M', 'J', 'J', 'A', 'S', 'O', 'N' };

2-D Array

One of the simplest form of multidimensional array is 2-D/Two dimensional array. It's expressed as an "array of arrays" or "array of one-dimensional array".

Two dimensional array stored in form of the row-column matrix, where the first index designates the row and second index denotes the column.

Example

data-type variable-name [size1][size2]

int monthly_day [12][30]

[12x30]

in $a[5][2] = \{ \{1, 2\}, \{2, 3\}, \{4, 5\}, \{6, 9\}, \{10, 15\} \}$

7b) An infinite loop is a looping construct that does not terminate the loop and execute the loop forever.

It's also called an indefinite loop or an endless loop.
It either produce a continuous output or no output.

Yes it's harmful for programming. More on internet

```
1  #include <iostream>
2  using namespace std;
3
4
5  class Rectangle {
6  private:
7      double length, width;
8
9  public:
10     void setData(double x, double y) {
11         length = x;
12         width = y;
13     }
14     double parameter() {
15         return 2.0 * (length + width);
16     }
17     double area() {
18         return length * width;
19     }
20     void showData() {
21         cout << "The Length of the rectangle is: " << length << endl;
22         cout << "The Width of the rectangle is: " << width << endl;
23         cout << "The Parameter of the rectangle is: " << parameter() << endl;
24         cout << "The Area of the rectangle is: " << area() << endl;
25 }
```

```
2019_7_c.cpp ✘ X
2019_7_c (Global Scope) m

25
26 }
27 }
28 };
29 int main() {
30
31     Rectangle ss;
32     double a, b;
33     cout << "Input the Length: ";
34     cin >> a;
35     cout << "Input the Width: ";
36     cin >> b;
37     ss.setData(a, b);
38     ss.showData();
39     return 0;
40
41 }
```

Q8a) An algorithm is a set of instruction for solving a problem or accomplishing a task.

One common example of an algorithm is a recipe which consists of specific instructions for preparing a dish or meal. Every computerized device uses algorithms to perform its functions in the form of hardware - or software-based routines.

Data types used in C++		Range
		bytes
int	integer	2 or 4
float	Floating point	4
double	Double floating point	8
char	character	1
wchar_t	wide character	2
bool	Boolean	1
void	Empty	0

```
1 #include <iostream>
2 using namespace std;
3 int main() {
4     int j = 0;
5     int rawscores[5] = {80,70,60,50,33};
6     char grades[5] = {'A', 'B', 'C', 'D', 'E'};
7     double rawscore;
8     long idNum = 0;
9     char grade;
10    while (cin >> idNum >> rawscore) {
11        for (int j = 4; j >= 0; j--)
12        {
13            if (rawscore > rawscores[j]) {
14                break;
15            }
16        }
17        grade = grades[j];
18        cout << idNum << " " << grade << endl;
19    }
20
21    return 0;
22 }
```

2019_8_c.cpp* ✘ X

2019_8_c (Global Scope) fun(int a, int & b)

```
1 #include <iostream>
2 using namespace std;
3
4 int fun(int a, int &b);
5
6 int main() {
7     int a = 1;
8     int b = 2;
9     int c = 3;
10    c = fun(a, b);
11
12    /* here c becomes 42 bcz the function returns 42
13       as the function takes address of b and its call by reference
14       so it swap the value of b with new value 42 */
15    cout << c << " " << b << " " << a << endl;
16
17    return 0;
18}
19 int fun(int a, int &b) {
20     a = 42;
21     b = 42; //call by reference and swap b=2 with b=42
22     return 42;
23}
24
```