C++ BASICS RELEVANT TO CSI 228

FARIHA TABASSUM ISLAM

BASICS

- C++ language extends the C programming language with additional features such as type checking, object-oriented programming, exception handling etc.
 - C++ was developed by Bjarne Stroustrup in 1979.
 - File extension .cpp
- Why C++ in this course?
 - The Standard Template Library (STL) of C++ provides useful codes
 - STL is a set of C++ template classes to provide common programming data structures and functions such as lists, stacks, arrays, etc. It is a library of container classes, algorithms, and iterators.

SIMILARITIES WITH C

- Variables, Operators
- struct
- Array
- Function
- Pointer
- Strings
- If, if...else-if statement, switch case, for loop, while loop, do-while loop, continue statement, break statement, goto statement
- Recursion

can use the same code as written in c

BASICS

```
C++

1 #include<iostream>
2 using namespace std;

3

4 int main()

5 {
    cout<< "Hello World!";
    return 0;

8 }</pre>
```

for details explanation: https://beginnersbook.com/2017/08/first-cpp-program/

```
1 #include<stdio.h>
2
3 int main()
4 {
5    printf("Hello World!");
6    return 0;
7 }
```

VARIABLES AND DATA TYPES

- int
- char
- bool
 - holds Boolean value true or false
- double
- float

```
#include <iostream>
                            Output:
    #include <cstdio>
                            inside first if
    using namespace std;
                            end
    int main()
       bool b1 = true;
       bool b2 = false;
       if (b1) {
           printf("inside first if\n");
10
11
           if (b2) {
               printf("inside first nested if\n");
12
13
14
       printf("end\n");
15
       return 0;
16
17
```

```
#include <iostream>
                           Output:
    #include <cstdio>
                           inside first if
    using namespace std;
                           inside third nested if
    int main()
                           end
 6
       bool b1 = true;
       bool b2 = false;
 8
       bool b3 = 0;
       bool b4 = 1;
10
       if (b1) {
11
           printf("inside first if\n");
12
13
           if (b2) {
                printf("inside first nested if\n");
14
15
16
           if (b3) {
                printf("inside second nested if\n");
17
18
19
           if (b4) {
               printf("inside third nested if\n");
20
21
22
       printf("end\n");
23
24
       return 0;
25
```

scanf(), printf() EQUIVALENT

```
C++
   #include <iostream>
   using namespace std;
   int main()
4
5
       int a;
       cin >> a;
6
       cout << a;
8
        return 0;
9
```

```
#include <stdio.h>
   int main()
3
       int a;
       scanf("%d",&a);
       printf("%d",a);
       return 0;
8
```

scanf(), printf() EQUIVALENT

```
1 #include <iostream>
                                             1 #include <stdio.h>
                            which one is
 2 using namespace std;
                                                int main()
                            the C++ code?
    int main()
                                             3
 4
                                                    char str[100];
                                             4
        char str[100];
                                                    /* input single word */
        /* input single word */
                                                    scanf("%s", str);
                                             6
        cin >> str;
                                                    printf("%s", str);
        cout << str;
                                                    /* discards the input buffer */
        /* discards the input buffer */
                                                    fflush(stdin);
        cin.sync();
10
                                                    /* input a line */
                                            10
       /* input a line */
11
                                                    fgets(str, 100, stdin);
                                           11
        cin.get(str, 100);
12
                                                    printf("%s", str);
                                           12
13
        cout << str;</pre>
                                           13
                                                    return 0;
14
        return 0;
                                           14
15
```

IF YOU PREFER scanf(), printf() OVER cin, cout

```
#include <iostream>
   #include <cstdio>
   using namespace std;
   int main()
        int a;
 6
        scanf("%d",&a);
        printf("%d",a);
        return 0;
10
```

STANDARD TEMPLATE LIBRARY (STL)

- STL is a set of C++ template classes to provide common programming data structures and functions such as lists, stacks, arrays, etc. It is a library of container classes, algorithms, and iterators.
- We are going to use
 - Sorting
 - vector
 - priority queue

VECTOR

- #include <vector>
- To know more about vectors: https://www.edureka.co/blog/vectors-in-cpp/

```
#include <iostream>
                                                19
                                                         /* iterate over vector - way 1 */
                                                         for (int i = 0; i < list4.size(); i++)
   #include <cstdio>
                                                20
                                                             printf("[%d] %d\n", i, list4[i]);
   #include <vector>
                                                21
                                                22
                                                         /* delete from index i */
    using namespace std;
                                                23
                                                         int i = 2:
    int main()
                                                         list4.erase(list4.begin() + i);
 6
                                                24
        /* initialization - way 1 */
                                                25
                                                         /* iterate over vector - way 2 */
        vector<int> list2 = {1, 10, 200};
                                                26
                                                         for (int x : list4)
 8
        /* initialization - way 2 */
                                                27
                                                             printf("%d\n", x);
 9
                                                28
10
        vector<int> list4;
11
        int x;
                                                Input:
                                                11 753 2 8 91
        for (int i = 0; i < 5; i++)
12
                                                Output:
13
                                                size: 5
14
            cin >> x;
                                                [0] 11
15
            list4.push_back(x);
                                                 [1] 753
16
                                                 [2] 2
                                                [3] 8
17
        /* size of vector */
                                                [4] 91
        printf("size: %d\n", list4.size());
18
                                                11
FARIHA TABASSUM ISLAM, LECTURER, DEPT. OF CSE, UIU
                                                753
```

8 91

```
19
                                                        /* iterate over vector - way 1 */
    #include <bits/stdc++.h>
                                                        for (int i = 0; i < list4.size(); i++)
                                               20
                                                            printf("[%d] %d\n", i, list4[i]);
                                               21
 3
                                               22
                                                        /* delete from index i */
    using namespace std;
                                               23
                                                        int i = 2:
    int main()
                                                        list4.erase(list4.begin() + i);
                                               24
 6
                                                        /* iterate over vector - way 2 */
        /* initialization - way 1 */
                                               25
        vector<int> list2 = {1, 10, 200};
                                               26
                                                        for (int x : list4)
 8
        /* initialization - way 2 */
                                               27
                                                            printf("%d\n", x);
 9
                                               28
10
        vector<int> list4;
11
        int x;
                                               Input:
                                               11 753 2 8 91
        for (int i = 0; i < 5; i++)
12
                                               Output:
13
                                               size: 5
14
            cin >> x;
                                                [0] 11
15
            list4.push_back(x);
                                                [1] 753
16
                                                [2] 2
                                               [3] 8
17
        /* size of vector */
                                               [4] 91
        printf("size: %d\n", list4.size());
18
                                               11
                                               753
```

8 91

SORTING

- sort array
- sort vector
- sort structure

SORT Array

```
1 #include <iostream>
   #include <cstdio>
    #include <bits/stdc++.h>
    using namespace std;
    int main()
 6
        int arr[] = \{100, 512, 6, 724, 31, 14, 2, 0\};
        /* Length of the array */
        int len = sizeof(arr) / sizeof(arr[0]);
 9
10
        /* print the array */
        for (int i = 0; i < len; i++)
11
12
            printf("%d ", arr[i]);
13
        printf("\n");
14
        /* sort the array */
15
        sort(arr, arr + len);
        /* print the array */
16
        for (int i = 0; i < len; i++)
17
            printf("%d ", arr[i]);
18
19
        printf("\n");
20
                                         Default order/
21
        return 0;
                                       Ascending order
22
```

Output:

100 512 6 724 31 14 2 0 0 2 6 14 31 100 512 724

SORT Array

```
#include <iostream>
1 #include <iostream>
                               Output:
                                                                                          Output:
                               100 512 6 724 31 14 2 0
                                                              #include <cstdio>
                                                                                          100 512 6 724 31 14 2 0
2 #include <cstdio>
                               0 2 6 14 31 100 512 724
                                                                                          724 512 100 31 14 6 2 0
    #include <bits/stdc++.h>
                                                               #include <bits/stdc++.h>
    using namespace std;
                                                               using namespace std;
    int main()
                                                               int main()
 6
        int arr[] = \{100, 512, 6, 724, 31, 14, 2, 0\};
                                                                   int arr[] = \{100, 512, 6, 724, 31, 14, 2, 0\};
        /* Length of the array */
                                                                   /* Length of the array */
        int len = sizeof(arr) / sizeof(arr[0]);
                                                                   int len = sizeof(arr) / sizeof(arr[0]);
 9
        /* print the array */
                                                                   /* print the array */
10
                                                           10
        for (int i = 0; i < len; i++)
                                                                   for (int i = 0; i < len; i++)
11
                                                           11
            printf("%d ", arr[i]);
12
                                                           12
                                                                        printf("%d ", arr[i]);
13
        printf("\n");
                                                           13
                                                                    printf("\n");
        /* sort the array */
14
                                                                   /* sort the array */
                                                           14
15
        sort(arr, arr + len);
                                                                   sort(arr, arr + len, greater<int>());
                                                           15
        /* print the array */
16
                                                                   /* print the array */
                                                           16
        for (int i = 0; i < len; i++)
17
                                                           17
                                                                    for (int i = 0; i < len; i++)
18
            printf("%d ", arr[i]);
                                                                        printf("%d ", arr[i]);
                                                           18
19
        printf("\n");
                                                           19
                                                                    printf("\n");
20
                                                           20
                                         Default order/
21
        return 0;
                                                           21
                                                                   return 0;
                                                                                                Descending order
                                       Ascending order
22
                                                           22
```

SORT Array of struct

```
#include <bits/stdc++.h>
                                                        Output:
    using namespace std;
                                                         a:5 b:5
    struct Pair
                                                        a:1 b:6
       int a, b;
                                                        a:3 b:9
 6
                                                        a:3 b:12
    bool comp(Pair p1, Pair p2)
                                                        a:8 b:16
 8
                                                        a:5 b:100
 9
        return p1.b < p2.b;
10
    int main()
12
       /* an array of struct */
13
14
        Pair arr[] = {{5, 100}, {3, 9}, {3, 12}, {1, 6}, {5, 5}, {8, 16}};
15
        int n = sizeof(arr) / sizeof(arr[0]);
       /* sort the array */
16
        sort(arr, arr + n, comp);
17
       /* print the array */
18
        for (int i = 0; i < n; i++)
19
20
           printf("a:%d b:%d\n",arr[i].a, arr[i].b);
21
22
23
                                          No default order. Order must
24
        return 0;
                                              be specified by a function
25
```

this function is a must for sorting an array of struct

SORT Array of struct

```
1 #include <bits/stdc++.h>
                                                    Output:
    using namespace std;
                                                    a:5 b:100 ratio:20
    struct Pair
                                                    a:1 b:6 ratio:6
       int a, b;
                                                    a:3 b:12 ratio:4
 6
                                                    a:3 b:9 ratio:3
    bool comp2(Pair p1, Pair p2)
                                                    a:8 b:16 ratio:2
                                                    a:5 b:5 ratio:1
 9
       return p1.b / p1.a > p2.b / p2.a;
10
11
    int main()
12
13
       /* an array of struct */
       Pair arr[] = {{5, 100}, {3, 9}, {3, 12}, {1, 6}, {5, 5}, {8, 16}};
14
15
       int n = sizeof(arr) / sizeof(arr[0]);
       /* sort the array */
16
       sort(arr, arr + n, comp2);
17
       /* print the array */
18
19
       for (int i = 0; i < n; i++)
20
           printf("a:%d b:%d ratio:%d\n",arr[i].a, arr[i].b, arr[i].b/arr[i].a);
21
22
23
                                                No default order. Order must
24
        return 0;
                                                    be specified by a function
25
```

REFERENCES

- https://beginnersbook.com/2017/08/c-plus-plus-tutorial-for-beginners/
- https://www.edureka.co/blog/vectors-in-cpp/