

Assessment sub
X

(https://swayam.gov.in)



(https://swayam.gov.in/nc_details/NPTEL)

navnathdeshmukh363@gmail.com ✓

NPTEL (https://swayam.gov.in/explorer?ncCode=NPTEL) » Programming in Modern C++ (course)

Course
outlineHow does an
NPTEL
online
course
work? ()

Week 0 ()

Week 1 ()

Week 2 ()

Week 3 ()

Week 4 ()

Week 5 ()

Week 6 ()

Week 7 ()

Week 8 ()

Week 9 ()

☐ Lecture 41 :
Input-Output:
File Handling
in C (unit?
unit=102&lesson=103)

☐ Lecture 42 :
Input-Output:
Streams in

Thank you for taking the Week 9 : Assignment 9.

Week 9 : Assignment 9

Your last recorded submission was on 2023-03-25, 01:14 Due date: 2023-03-29, 23:59 IST.
IST

1) Consider the program given below. 2 points

```
#include <iostream>
#include <iomanip>
using namespace std;

int main () {
    int i = 65;
    cout << setbase(8)<< i << " ";
    cout << setbase(10)<< i << " ";
    cout << setbase(16)<< i << " ";
    cout << static_cast<char>(i) << endl;
    return 0;
}
```

What will be the output?

- a) 65 65 65 65
- b) 65 65 65 A
- c) 101 65 41 65
- d) 101 65 41 A

- ☐ a.
- ☐ b.

C++ (unit?
Assessment submitted
unit=102&lesson=104)
X

☐ Lecture 43 :
C++ Standard
Library: Part 1
(Generic
Programming)
(unit?
unit=102&lesson=105)

☐ Lecture 44 :
C++ Standard
Library: Part 2
(STL) (unit?
unit=102&lesson=106)

☐ Lecture 45 :
C++ Standard
Library: Part 3
(STL) (unit?
unit=102&lesson=107)

☐ Tutorial 09 :
How to design
a UDT like
built-in types?:
Part 3:
Updates and
Mixes of UDTs
(unit?
unit=102&lesson=108)

☐ Week 9
Lecture
Material (unit?
unit=102&lesson=109)

☒ **Quiz: Week 9
: Assignment
9
(assessment?
name=202)**

☐ W9_Programming_Qs-
1
(/noc23_cs50/progassignment?
name=204)

☐ W9_Programming_Qs-
2
(/noc23_cs50/progassignment?
name=205)

☐ W9_Programming_Qs-
3
(/noc23_cs50/progassignment?
name=207)

- ☐ c.
☒ d.

2)

Consider the program given below.

```
#include<cstdio>
using namespace std;
int main(){
    FILE *infp, *outfp;
    int c;
    if((infp = fopen("in.txt", "r")) == NULL)
        return -1;
    if((outfp = fopen("out.txt", "w")) == NULL)
        return -2;
    while((c = fgetc(infp)) != EOF && c != '\n')
        fputc(c, outfp);
    fclose(infp);
    fclose(outfp);
    return 0;
}
```

Choose the statement that is true about the output of the program.

- a) It copies the entire content file in.txt to file out.txt.
b) It copies the content file in.txt to file out.txt without the newlines (i.e. the entire content of in.txt would be copied to a single line of out.txt).
c) It copies the first line of file in.txt to file out.txt.
d) It copies the content file in.txt to file out.txt if file in.txt has only one line; otherwise, it generates an error in runtime.

- ☐ a.
☒ b.
☐ c.
☐ d.

3)

2 points

2 points

Assessment Submitted.

X

**Download
Videos ()****Books ()****Transcripts ()****Problem
Solving
Session ()**

Consider the following code segment.

```
#include <iostream>
#include <fstream>

int main () {
    std::ifstream infile("input.txt");
    std::string line;
    if (_____) { //LINE-1
        std::cout << "file does not exists";
    }
    else{
        while (getline(infile, line))
            std::cout << line << std::endl;
        infile.close();
    }
    return 0;
}
```

Choose the appropriate option to fill in the blank at LINE-1 such that it checks if the file input.txt does not exist.

- a) `infile.is_open()`
- b) `!infile.is_open()`
- c) `!infile.open()`
- d) `fopen(infile) == NULL`

- ☐ a.
- ☐ b.
- ☐ c.
- ☐ d.

4)

2 points

Assessment submitted.

X

Consider the following code segment.

```
#include<iostream>

template<class Itr, class T>
int findmax(Itr first, Itr last, T& mval) {
    int maxpos = 0, i = 0;
    mval = *first++;
    while (first != last) {
        if(*first > mval){
            mval = *first;
            maxpos = i + 1;
        }
        ++first;
        ++i;
    }
    return maxpos;
}

int main(){
    int iArr[] = { 3, 2, 6, 1, 6, 8, 7};
    double mVal = 0.0;

    -----; //LINE-1
    std::cout << pos << ", " << mVal;
    return 0;
}
```

Choose the appropriate options to fill in the blank at LINE-1 such that the program finds out the maximum element of the array iArr and the output is 5, 8.

- a) findmax(iArr, iArr + sizeof(iArr) / sizeof(*iArr), mVal)
- b) int pos = findmax(iArr, &iArr[sizeof(iArr) / sizeof(*iArr)], mVal)
- c) int mVal = findmax(iArr, iArr + sizeof(iArr) / sizeof(*iArr), mVal)
- d) int pos = findmax(iArr, iArr + sizeof(iArr) / sizeof(*iArr), mVal)

- ☐ a.
- ☒ b.
- ☐ c.
- ☒ d.

5)

2 points

Assessment submitted.

X

Consider the following code segment.

```
#include <iostream>
#include <iomanip>
using namespace std;
int main () {
    cout << setprecision(5) << setfill('0') << setw(10) << 10/3.0;
    return 0;
}
```

What will be the output?

- a) 00003.33333
- b) 00003.3333
- c) 00000000003.33333
- d) 00000000003.00000

- ☐ a.
- ☒ b.
- ☐ c.
- ☐ d.

6)

2 points

Assessment submitted.

X

Consider the following code segment (in C++11).

```
#include <iostream>
#include <algorithm>
#include <vector>
#include <list>

int main() {
    std::list<int> li= { 1, 2, 3, 4, 5, 6, 7, 8, 9 };
    std::vector<int> vi(li.size());
    std::list<int>::iterator it1 = li.begin();
    std::vector<int>::iterator it2 = vi.begin();
    for(int i = 0; i < 5; i++){    //LINE-1
        it1++; it2++;
    }

    copy(it1, li.end(), it2);

    for(it2 = vi.begin(); it2 != vi.end(); ++it2)
        std::cout << *it2;
    return 0;
}
```

What will be the output?

- a) 123456789
- b) 6789000000
- c) 000006789
- d) 000012345

- ☐ a.
- ☐ b.
- ☒ c.
- ☐ d.

7)

2 points

Assessment submitted.

X

Consider the following program (in C++11) to compute the inner product between the integers in a vector `vi` and a list `li`.

```
#include <iostream>
#include <list>
#include <vector>
#include <numeric>

int operation1(int i, int j){ return i + j; }
int operation2(int i, int j){ return i * j; }

int main() {
    std::vector<int> vi { 1, 2, 3, 4, 5 };
    std::list<int> li { 50, 40, 30, 20, 10 };

    int n = inner_product(_____, //LINE-1
        std::cout << n;
        return 0;
}
```

Choose the correct option to fill in the blank at LINE-1 so that output becomes 350.

- a) `vi.begin(), vi.end(), li.begin(), 0, operation1, operation2`
- b) `li.begin(), li.end(), vi.begin(), 0, operation1, operation2`
- c) `li.begin(), li.end(), vi.begin(), 0, operation2, operation1`
- d) `vi.begin(), vi.end(), li.begin(), 0, operation2, operation1`

- ☒ a.
- ☒ b.
- ☐ c.
- ☐ d.

8)

2 points

Assessment submitted.

X

Consider the following code segment (in C++11).

```
#include<iostream>
#include<list>

struct divisible{
    int d_;
    divisible(int d = 1) : d_(d) { }
    bool operator()(int i){ return (i % d_ == 0); }
};

template<class T, class P>
T find_if(T first, T last, P pred) {
    while (_____) ++first;    //LINE-1
    return first;
}

void print(std::list<int> li, int d){
    divisible divi(d);
    std::list<int>::iterator it = find_if(li.begin(), li.end(), divi);    //LINE-3
    while(it != li.end()){
        std::cout << *it << " ";
        it = find_if(++it, li.end(), divi);
    }
}

int main(){
    std::list<int> li {7, 8, 1, 4, 2, 5, 6, 3};
    int d;
    print(li, 4);
    return 0;
}
```

Choose the appropriate option to fill in the balnk at LINE-1 so that it prints the values from list li which are divisible by 4. So the output should be 8 4

- a) first != last || !pred(*first)
- b) first != last && !pred(*first)
- c) first != last && pred(*first)
- d) first != last || pred(first)

- ☐ a.
- ☒ b.
- ☐ c.
- ☐ d.

9)

2 points

Assessment submitted.

X

Consider the following code segment (in C++11).

```
#include<iostream>
#include<algorithm>
#include<vector>

class student{
public:
    student(int roll, char grade) : roll_(roll), grade_(grade){}
    int get_roll(){ return roll_; }
    char get_grade(){ return grade_; }
private:
    int roll_;
    char grade_;
};

struct comparator{
    bool operator()(student s1, student s2){
        if(s1.get_grade() == s2.get_grade())
            return s1.get_roll() < s2.get_roll();
        return s1.get_grade() > s2.get_grade();
    }
};

int main() {
    student s[] = { student(30, 'A'), student(10, 'B'),
                    student(20, 'C'), student(40, 'B') };
    std::vector<student> s_list(s, s + sizeof(s) / sizeof(*s));
    std::sort(s_list.begin(), s_list.end(), comparator());
    for(std::vector<student>::iterator it = s_list.begin(); it != s_list.end(); it++)
        std::cout << it->get_roll() << " : " << it->get_grade() << std::endl;
    return 0;
}
```

What will be the output?

- a) 30 : A
10 : B
40 : B
20 : C
- b) 20 : C
10 : B
40 : B
30 : A
- c) 20 : C
40 : B
10 : B
30 : A
- d) 30 : A
40 : B
10 : B
20 : C

- ☐ a.
- ☒ b.
- ☐ c.
- ☐ d.

You may submit any number of times before the due date. The final submission will be considered for grading.

Submit Answers

Assessment submitted.

X