Assingment 4

19CS10060

Sunanda Mandal

1 (a) Ans: (i) A technique of C++ that allows us to write inline functions without a name

(b) Ans: (i) The return type of lambda expression can be neglected in some cases

(c) Ans: (iii) 3

(d)

```
# include <iostream>
# include <vector>
# include <algorithm>
using namespace std;
int main() {
  vector<int> arr = {1, 2, 3, 6, 4, 9};
  sort(arr.begin(), arr.end(), [](int &a, int &b){
                                  return a > b; });
  return 0;
}
```

(e)

```
# include <iostream>
# include <vector>
# include <algorithm>
using namespace std;
int main(){
  vector<int> arr = {1, 2, 3, 6, 4, 9};
  int countEven = (arr.begin(), arr.end(), [](int&a){
                                return a%2==0;});

  return 0;
}
```

19CS10060

Sunanda Mandal

1. (f)

```
# include <iostream>

int main() {
    int n = 1024;
    auto modify = [&] () {
        // do some modification to n. for example :
            n /= 2;
    }
    modify ();
    return 0;
}
```

2. (a)

```
# include <iostream>
using namespace std;
class TOH {
  public:
    void operation ()(int n , char from_r, char to_r, char
                                          aux_r){
        if (n = =0)
            return;
        (*this) (n-1, from_r, aux_r, to_r);
        cout << "Move disk" << n <<"from stack"<<from_r<<
            "to stack" << to_r << endl;
        (*this) (n-1, aux_r, to_r, from_r);
    }
};
```

continue . . →

Sunanda Mandal 19CS10060

```
    int main() {
        int num_disk;
        cin >> num_disk;
        TOH tower;
        tower (num_disk, 'A', 'B', 'C');
        retur 0;
    }
```

2.(b)

```
    # include <iostream>
    # include<functional>
    using namespace std;
    int main() {
        function <void(int, char, char, char) > hanoi;
        hanoi = [&hanoi] (int n, char from_r, char to_r,
                               char aux_r) {
            if(n ==0)
                return;
            hanoi ((n-1), from_r, aux_r, to_r);
            cout << "Move disk" << n << "from stack" << from_r <<
                "to stack" << to_r << endl;
            hanoi (n-1, aux_r, to_r, from_r);
        };
        int num_disk;
        cin >> num_disk;
        hanoi(num_disk, 'A', 'B', 'C');
        return 0;
    }
```

3 (a) functons

```
# include < iostream >
# include < string >
# include <algorithm>
using namespace std;
class Permute {
    public :
      void operator() (string word, int l, int r){
          if (l== r)
              cout << word < endl;
          else {
              for (int i = l; i <=r; i++) {
                  swap (word [l], word[i] );
                  (*this) (word , l+1, r);
                  swap ( word[l], word[i] );
              }
          }
      }
};
int main() {
    Permute fcall;
    string s
    cin >> s ;
    fcall (s, 0, s.size()-1) ;
    return 0;
}
```

Suranda Mandal

19CS10060

3. (b)

```
# include <iostream>
# include <string >
# include <functional >
using namespace std;
int main(){
    string s;
    cin >> s;
    function void(int,int) permute = [&] (int l, int r){
        if (l==r)
            cout << s<< endl;
        else{
          for(int i =l; i<=r; i++){
              swap(s[l], s[i]);
              permute (l+1, r);
              swap(s[l], s[i]);
          }
        }
    };
    permute( 0, s.siz()-1);
    return 0;
}
```