

CS101 – Computer Programming
Quiz for Thursday Batch – 16 October 2014

Q1. Consider the following code fragment and choose the correct option(s) from below.

```
void swap(int *a, int *b) {
    int *temp;
    temp = a;
    a = b;
    b = temp;
}

int main() {
    int A = 4, B = 6;
    swap(&A, &B);
    cout << A << B;
    return 0;
}
```

- A) The program prints 46
- B) The program prints 64
- C) The swap function swaps the addresses of A and B
- D) The swap function swaps the values of a and b
- E) None of these

Q2. The below given program first finds the length of the main string entered by the user. It then asks for the substring to be searched in the main string. If the entered substring is found, it displays the message "substring found at start index 'startindx' and end index 'endindx'". After which, it asks for the substring to be replaced by the replacement substring, both are to be entered by the user. Finally, it prints the main string.

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

//gets the string entered by the user of length 'len'
//It also checks for the maximum number of elements 'maxlength' the string can contain
//str[] is null-terminated at the end
void getstring(char str[], int maxlength, int &len){
    len=0;
    char c;
    while((c=getchar())!='\n'){
        if(len==maxlength-1){
            cout<<"Input exceeded"<<len<<endl;
            break;
        }
        else{
            str[len]=c;
            len++;
        }
    }
    str[len]='\0';
}

//Identifies the start and the end index of the word present in the main string
bool findsubstring(char string[], char word[], int l1, int l2, int &startindex, int &endindex){
    bool flag=false; int count, k, i, j;
    for(i=0; i<=l1-l2; i++){
        count=0;
        k=i;
        for(j=0; j<l2; j++){
            if(string[k]==word[j]) count++; k++;
        }
        if(count==l2){startindex=i; endindex=i+count-1; flag = true;}
    }
    return flag;
}
```

```

//finds the old string and replaces with the new string in the main string
void findandreplace(char A[], int len1,char oldstring[], char newstring[], int len2){
    for(int i=0;i<len1;i++)
        for(int j=0;i<len2;i++)
            if(A[i] == oldstring[j]) A[i]= newstring[j];
}

//prints the string on the screen
void printstring(char A[], int len){
    for(int i=0;i<len;i++) cout<<A[i];
    cout<<endl;
}

int main(){
    char c;int n1=1000, n2=100;
    char str[n1], word[n2];
    int l1,l2;
    int startindx, endindx;
    bool foundstring=false;

    cout<<"Enter the string of length <1000:";
    getstring(str,n1,l1);
    cout<<"Enter the substring to be searched of length <100:";
    getstring(word,n2,l2);
    cout<<"l1:"<<l1<<" " <<"l2:"<<l2<<endl;

    foundstring=findsubstring(str,word,l1,l2,startindx, endindx);
    if(foundstring)
        cout<<"substring found at start index " <<startindx <<" and end index " << endindx
        <<endl;
    else
        cout<<"substring not found";

    char oldstr[n2],newstr[n2]; int l3;
    cout<<"Enter a substring to be replaced:"<<endl;
    getstring(oldstr,n2,l3);
    cout<<"Enter a replacement substring:"<<endl;
    getstring(newstr,n2,l3);

    findandreplace(str,l1,oldstr, newstr, l3);
    cout<<"Printing new string"<<endl;
    printstring(str,l1);
    return 0;
}

```

Which are the appropriate choices for the above program

- A) The required substring cannot be found, but, it will find the substring, if the statement "if(count==l2){startindex=i; endindex=i+count-1; flag = true;}" in the above findsubstring function is replaced by "if(count==l2){startindex=i; endindex=j; flag = true;}"
- B) The findandreplace function does not do its job properly
- C) The function getstring does not get the proper length of the string and hence leads to problems in searching substring
- D) The start index and the end index calculated for substring search is not correct
- E) None of these