You can view this report online at :

| | |
|---|---|
| Full Name: | |
| Email: | |
| Test Name: | **CS224 - Term Exam I - T2 - Fall 2023** |
| Taken On: | 22 Sep 2023 09:21:33 PKT |
| Time Taken: | 161 min 12 sec/ 140 min |
| Work Experience: | < 1 years |
| Invited by: | Munzir |
| Skills Score: | |
| Tags Score: | CS224 45/50<br>pass by pointe 0/50<br>pass by reference 0/50 |

**43.6%**

**120/275**

scored in **CS224 - Term Exam I - T2 - Fall 2023** in 161 min 12 sec on 22 Sep 2023 09:21:33 PKT

**Recruiter/Team Comments:**

*No Comments.*

| | Question Description | Time Taken | Score | Status |
|---|---|---|---|---|
| Q1 | Word Sorting Game > **Coding** | 41 min 3 sec | 50/ 50 | ✓ |
| Q2 | The Karachi Currency Exchange > **Coding** | 22 min 49 sec | 45/ 50 | ✓ |
| Q3 | Date and Time > **Coding** | 55 min 6 sec | 25/ 125 | ✓ |
| Q4 | Prepare Biryani > **Coding** | 41 min 43 sec | 0/ 50 | ✗ |

**QUESTION 1**

Correct Answer

Score 50

## Word Sorting Game > Coding

QUESTION DESCRIPTION

You are tasked with creating a word sorting game. The game starts with a list of words, and your task is to arrange them in alphabetical order. However, there's a twist: you are only allowed to use dynamically allocated arrays of character arrays (strings), and you should implement the sorting algorithm yourself. You cannot use classes, structs, strings, or vectors.

Write a C++ program that takes a list of words as input and sorts them alphabetically using the following rules:

1. The words are case-insensitive, meaning "apple" and "Apple" should be treated as the same word.
2. Words starting with uppercase letters should come before words starting with lowercase letters in the sorted list.

Your program should follow these steps:

1. Dynamically allocate memory for an array of character arrays (strings) to store the words.

2. Read words from the standard input until the user enters "END" (without quotes) to signal the end of input.
3. Sort the words according to the rules mentioned above.
4. Display the sorted list of words, one word per line, to the standard output.

**Example Input and Output:**

Input:

```
banana
Apple
cherry
Dog
END
```

Output:

```
Apple
Dog
banana
cherry
```

**CANDIDATE ANSWER**

Language used: **C++14**

```
#include <map>
#include <set>
#include <list>
#include <cmath>
#include <ctime>
#include <deque>
#include <queue>
#include <stack>
#include <string>
#include <bitset>
#include <cstdio>
#include <limits>
#include <vector>
#include <climits>
#include <cstring>
#include <cstdlib>
#include <fstream>
#include <numeric>
#include <sstream>
#include <iostream>
#include <algorithm>
#include <unordered_map>

using namespace std;
void sort(char** arr,int size)
{
    for (int j = 0;j<size-1;j++)
    {
        for (int i=0; i<size-j-1;i++)
        {
            int index = 0;
            while (arr[i][index] != '\0')
            {
                if (arr[i][index] < arr[i+1][index])
                {
```

```
                    break;
                }
                else if (arr[i][index]>arr[i+1][index])
                {
                    char*temp = arr[i];
                    arr[i] = arr[i+1];
                    arr[i+1] = temp;
                    break;
                }
                index++;
            }
        }
    }
}
int main()
{
    /* Enter your code here. Read input from STDIN. Print output to STDOUT */
    int size = 10;
    char **names= new char*[size];

    for (int i=0; i<size ; i++) //loop to iterate and take inputs
    {
        char *c = new char[50];
        cin.getline(c,50);
        if (c[0] == 'E'&&c[1] == 'N'&&c[2]=='D')
        {
            size = i;
            break;
        }
        names[i] = c;

    }

    sort(names,size);
    //now printing results
    for (int i = 0; i<size; i++)
    {
        for (int x=0;x<50;x++)
        {
            if (names[i][x] == '\0')
            {
                break;
            }
            cout<<names[i][x];
        }
        cout<<"\n";
    }

    return 0;
}
```

| TESTCASE | DIFFICULTY | TYPE | STATUS | SCORE | TIME TAKEN | MEMORY USED |
|---|---|---|---|---|---|---|
| Testcase 0 | Easy | Sample case | Success | 10 | 0.0235 sec | 8.82 KB |
| Testcase 1 | Easy | Hidden case | Success | 10 | 0.028 sec | 8.89 KB |
| Testcase 2 | Easy | Hidden case | Success | 10 | 0.0365 sec | 8.78 KB |
| Testcase 3 | Easy | Hidden case | Success | 10 | 0.0241 sec | 8.77 KB |
| Testcase 4 | Easy | Sample case | Success | 10 | 0.0302 sec | 8.8 KB |

No Comments

QUESTION 2

Correct Answer

Score 45

## The Karachi Currency Exchange › Coding CS224

QUESTION DESCRIPTION

In Karachi, the currency exchange market is as unpredictable as Karachi's weather! Write a C++ function to help a savvy trader maximize their profit by buying and selling Pakistani Rupees on different days. Remember, in Karachi, even the Rupees have mood swings!

Your task is to design a C++ function `maximizeProfit` that takes an array of daily currency prices as input and returns the maximum profit that can be made by buying and selling Rupees on different days.

The function should have the following signature:

```
int maximizeProfit(int prices[], int n);
```

**Input:**

- An array `prices` containing `n` integers where `prices[i]` represents the price of the Rupee on day `i`.
- The integer `n` (1 <= n <= 10^5), representing the number of days.

**Output:**

- An integer representing the maximum profit that can be made by buying and selling Rupees on different days.

**Constraints:**

- You can only buy and sell Rupees once per day.
- Short selling (selling before buying) is not allowed.
- The goal is to maximize profit, so you should find the best buying and selling days.

INTERVIEWER GUIDELINES

```
#include <iostream>
#include <climits>

using namespace std;

int maximizeProfit(int prices[], int n) {
    int maxProfit = 0;
    int minPrice = INT_MAX;

    for (int i = 0; i < n; ++i) {
        // Update the minimum price encountered so far
        minPrice = min(minPrice, prices[i]);

        // Calculate the profit that can be made if we sell on the
current day
        int currentProfit = prices[i] - minPrice;

        // Update the maximum profit if the current profit is greater
        maxProfit = max(maxProfit, currentProfit);
    }

    return maxProfit;
}

int main() {
    int n;
    cin >> n;

    int prices[n];
    for (int i = 0; i < n; ++i) {
        cin >> prices[i];
    }
```

```
    int result = maximizeProfit(prices, n);
    cout << result << endl;

    return 0;
}
```

**CANDIDATE ANSWER**

Language used: **C++20**

```
#include <map>
#include <set>
#include <list>
#include <cmath>
#include <ctime>
#include <deque>
#include <queue>
#include <stack>
#include <string>
#include <bitset>
#include <cstdio>
#include <limits>
#include <vector>
#include <climits>
#include <cstring>
#include <cstdlib>
#include <fstream>
#include <numeric>
#include <sstream>
#include <iostream>
#include <algorithm>
#include <unordered_map>

using namespace std;

int maximizeprofit(int prices[], int n)
{
    int *minvalue = &prices[0];
    int *maxvalue = &prices[0];

    for (int i = 0 ; i<n ; i++)
    {
        if (prices[i] < *minvalue)
        {
            minvalue = &prices[i];
            maxvalue = &prices[i];
        }
        else if (prices[i] > *maxvalue)
        {
            maxvalue = &prices[i];
        }
    }
    //cout<< *maxvalue<<endl;
    //cout<< *minvalue<<endl;
    return *maxvalue - *minvalue;
}

int main() {
    /* Enter your code here. Read input from STDIN. Print output to STDOUT */
    int Totaldays ;
    cin >>Totaldays;
```

```
    int *prices = new int[Totaldays];

    for (int i =0 ; i < Totaldays ; i++)
    {
        cin >> prices[i];
    }
    cout <<maximizeprofit(prices,Totaldays);

    }
```

| TESTCASE | DIFFICULTY | TYPE | STATUS | SCORE | TIME TAKEN | MEMORY USED |
|---|---|---|---|---|---|---|
| Testcase 0 | Easy | Sample case | Success | 5 | 0.0664 sec | 8.86 KB |
| Testcase 1 | Easy | Sample case | Success | 5 | 0.033 sec | 8.86 KB |
| Testcase 2 | Easy | Hidden case | Success | 5 | 0.0689 sec | 8.89 KB |
| Testcase 3 | Easy | Hidden case | Success | 5 | 0.1147 sec | 8.79 KB |
| Testcase 4 | Easy | Hidden case | Success | 5 | 0.0525 sec | 8.77 KB |
| Testcase 5 | Easy | Hidden case | Success | 5 | 0.0981 sec | 8.84 KB |
| Testcase 6 | Easy | Hidden case | Success | 5 | 0.0353 sec | 8.8 KB |
| Testcase 7 | Easy | Hidden case | Wrong Answer | 0 | 0.0437 sec | 8.88 KB |
| Testcase 8 | Easy | Hidden case | Success | 5 | 0.0674 sec | 8.79 KB |
| Testcase 9 | Easy | Hidden case | Success | 5 | 0.0257 sec | 8.81 KB |

No Comments

**QUESTION 3**

Correct Answer

Score 25

## Date and Time › Coding

QUESTION DESCRIPTION

You are tasked with designing a C++ class named `DateTime` that represents a date and time. The class should have the following features:

1. The class should store the date in the format `dd/mm/yyyy` and the time in the format `hh:mm:ss`.
2. The class should provide methods to:
   a. Set and retrieve the date and time.
   b. Calculate the difference in days between two `DateTime` objects.
   c. Check if a given date is a public holiday in Karachi, Pakistan.
3. Implement a constructor that initializes the `DateTime` object with the current date and time.

You also need to write a C++ program that utilizes the `DateTime` class. Implement the following in your program:

- Prompt the user for input to create two `DateTime` objects and calculate the difference in days between them.
- Displays whether the provided dates are public holidays

Assume the following public holidays in Karachi, Pakistan:

- 14th August (Independence Day)
- 25th December (Quaid-e-Azam Day)

Your program should provide the user with the appropriate messages for each case.

Here's the skeleton code for the `DateTime` class:

```
#include <iostream>
#include <string>
```

```
class DateTime {
private:
    // Add private members for date and time storage

public:
    // Constructor to initialize with current date and time

    // Methods to set and retrieve date and time

    // Method to calculate difference in days

    // Method to check if a date is a public holiday in Karachi, Pakistan
};

// Define public holidays as constants

int main() {
    // Implement the main function to create DateTime objects, calculate
the difference, and check holidays

    return 0;
}
```

**CANDIDATE ANSWER**

Language used: **C++**

```
#include <iostream>
#include <string>

class DateTime {
private:
    int day,month,year,hour,minute,second;

public:
//Construct as indeppendence day
    DateTime(int d = 0,int mo = 0,int y = 0,int h = 0,int m = 0,int s = 0)
    {
        day = d;
        month = mo;
        year = y;
        hour = h;
        minute = m;
        second = s;
    }

    void SetDateTime(int d,int mo, int y, int h,int m,int s)
    {
        day = d;
        month = mo;
        year = y;
        hour= h;
        minute = m;
        second = s;
    }

    int CalculateDaysDifference(DateTime Comp_obj)
    {
        int d = 0;
        int mo = 0;
        int y = 0;
        d += day -Comp_obj.day;
        mo += month - Comp_obj.month;
```

```
        y += year - Comp_obj.year;

        int TotalDays = 0;
        TotalDays += d;
        TotalDays += mo * 30;//avg days
        TotalDays += y*365;

        return abs(TotalDays);
    }

    void ispublicholiday(DateTime Ind,DateTime QD)
    {
        char day0 = '\0';
        char month0 = '\0';
        if (day <10)
            day0 ='0';
        if (month <10)
            month0 ='0';


        if ((Ind.day == day && Ind.month == month) || (QD.month == month &&
QD.day == day))
        {
            std::cout << day0<<day <<" "<< month0<<month << " "<< year <<" is
a public Holiday"<<std::endl;
        }
        else
        {
            std::cout << day0<<day <<" "<<month0<<month <<" "<< year <<" is
not a public holiday"<<std::endl;
        }
    }

};

// Define public holidays as constants
const DateTime IndependenceDay(14, 8, 0, 0, 0, 0);
const DateTime QuaidEAzamDay(25, 12, 0, 0, 0, 0);

int main() {
    DateTime dt1, dt2;
    int day, month, year, hour, minute, second;

    //std::cout << "Enter date and time for DateTime object 1 (dd mm yyyy hh
mm ss): ";
    std::cin >> day >> month >> year >> hour >> minute >> second;
    dt1.SetDateTime(day, month, year, hour, minute, second);

    //std::cout << "Enter date and time for DateTime object 2 (dd mm yyyy hh
mm ss): ";
    std::cin >> day >> month >> year >> hour >> minute >> second;
    dt2.SetDateTime(day, month, year, hour, minute, second);

    int daysDifference = dt1.CalculateDaysDifference(dt2);
    std::cout << "Difference in days: " << daysDifference << std::endl;

    dt1.ispublicholiday(IndependenceDay,QuaidEAzamDay);
    dt2.ispublicholiday(IndependenceDay,QuaidEAzamDay);
}
```

| TESTCASE | DIFFICULTY | TYPE | STATUS | SCORE | TIME TAKEN | MEMORY USED |
|---|---|---|---|---|---|---|
| Testcase 0 | Easy | Sample case | Success | 25 | 0.0574 sec | 8.85 KB |

| | | | | | | |
|---|---|---|---|---|---|---|
| Testcase 1 | Easy | Sample case | Wrong Answer | 0 | 0.0263 sec | 8.81 KB |
| Testcase 2 | Easy | Hidden case | Wrong Answer | 0 | 0.057 sec | 8.95 KB |
| Testcase 3 | Easy | Hidden case | Wrong Answer | 0 | 0.0465 sec | 8.91 KB |
| Testcase 4 | Easy | Hidden case | Wrong Answer | 0 | 0.0259 sec | 8.75 KB |

No Comments

QUESTION 4

Wrong Answer

Score 0

## Prepare Biryani › Coding

pass by reference | pass by pointe

QUESTION DESCRIPTION

Karachi is known for its diverse cuisine, and one popular dish is "Biryani." However, not everyone in Karachi likes their Biryani the same way. Some like it spicy, some mild, and some with extra meat. You are tasked with writing a C++ program that helps a Biryani vendor prepare Biryani orders based on customer preferences.

You are given the following information:

- The vendor has a set recipe for Biryani, which includes a fixed amount of rice, spices, and meat.
- The vendor charges a fixed base price of 100 PKR for Biryani
- Customers can customize their orders with the following preferences:
  - Spice Level (int):
    - 1 for mild (10 PKR)
    - 2 for medium (20 PKR)
    - 3 for spicy (30 PKR)
  - Extra Meat (bool):
    - 1 for extra meat (50 PKR)
    - 0 for regular meat (0 PKR)
- Each order consists of spice level and extra meat preferences.

You need to implement a C++ program with the following:

1. Define a function `prepareBiryani` that takes the following parameters:
   - `int spiceLevel` passed by value.
   - `bool extraMeat` passed by reference.
   - spiceLevelString character array
   - `double totalCost` passed by pointer.

   The function should update `totalCost` based on the preferences and and the spiceLevelString indicating the spice level (e.g., "Mild," "Medium," or "Spicy").
2. In the `main` function, read the input values for spice level and extra meat preference.
3. Call the `prepareBiryani` function to calculate the total cost and update the spice level string.
4. Display the spice level and total cost of the Biryani.

**Input:**

- The input consists of the following values:
  - `spiceLevel` (1 <= spiceLevel <= 3) - An integer representing the spice level preference.
  - `extraMeat` (true or false) - A boolean representing the extra meat preference.

**Output:**

- The program should print the spice level (e.g., "Mild," "Medium," or "Spicy") and the total cost of the Biryani.

INTERVIEWER GUIDELINES

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

void prepareBiryani(int spiceLevel, bool& extraMeat, char*
spiceLevelString, double* totalCost) {

    if (spiceLevel == 1) {
        strcpy(spiceLevelString, "Mild");
        *totalCost = 100;  // Base price
    } else if (spiceLevel == 2) {
        strcpy(spiceLevelString, "Medium");
```

```
        *totalCost = 120;  // Base price + medium spice cost
    } else if (spiceLevel == 3) {
        strcpy(spiceLevelString, "Spicy");
        *totalCost = 150;  // Base price + spicy spice cost
    }

    if (extraMeat) {
        *totalCost += 50;  // Extra meat cost
    }
}

int main() {
    int spiceLevel;
    bool extraMeat;
    double totalCost = 0;
    char spiceLevelString[10];

    cin >> spiceLevel >> extraMeat;

    prepareBiryani(spiceLevel, extraMeat, spiceLevelString, &totalCost);

    cout << "Spice Level: " << spiceLevelString << endl;
    cout << "Total Cost: " << totalCost << " PKR" << endl;

    return 0;
}
```

**CANDIDATE ANSWER**

Language used: **C++14**

```
#include <map>
#include <set>
#include <list>
#include <cmath>
#include <ctime>
#include <deque>
#include <queue>
#include <stack>
#include <string>
#include <bitset>
#include <cstdio>
#include <limits>
#include <vector>
#include <climits>
#include <cstring>
#include <cstdlib>
#include <fstream>
#include <numeric>
#include <sstream>
#include <iostream>
#include <algorithm>
#include <unordered_map>

using namespace std;
void preparebiryani(int s,bool&m,char*,double *t);

int main() {
    /* Enter your code here. Read input from STDIN. Print output to STDOUT */

    char *c = new char [10];
    double *totalcost = new double;
    int spicelevel;
```

```
    int meeat;
    bool meat;

    cin >>spicelevel;
    cin >>meeat;
    if (meeat == 1)
    {
        meat =true;
    }
    else meat=false;

    preparebiryani(spicelevel, meat,c,totalcost);

    cout<< "Spice Level: ";
    int i=0;
    while(c[i] != '\0')
    {
        cout<<c[i];
    }
    cout<<"\n"<<"Total Cost is: "<<totalcost;
    return 0;
}

void prepareBiryani(int spicelvl, bool &meat, char *spicelev, double
*totalcost)
{


    if (meat)
    {
        *totalcost += 50;
    }

    if (spicelvl==1)
    {
        *totalcost += 10;
        spicelev[0] = 'M';
        spicelev[1] = 'I';
        spicelev[2] = 'L';
        spicelev[3] = 'D';
        spicelev[4] = '\0';
    }

    else if (spicelvl == 2)
    {

        *totalcost += 20;
        spicelev[0] = 'M';
        spicelev[1] ='E';
        spicelev[2] ='D';
        spicelev[3] = 'I';
        spicelev[4] = 'U';
        spicelev[5] = 'M';
        spicelev[6] = '\0';


    }
    else if (spicelvl==3)
    {
        *totalcost += 30;
        spicelev[0] ='S';
        spicelev[1] = 'p';
        spicelev[2] = 'i';
```

```
        spicelev[3] = 'c';
        spicelev[4] = 'y';
        spicelev[5] = '\0';
    }

}

```

**Result:** Compilation Failed

Compile Message

```
/usr/bin/ld: ./ccjZWahH.o: in function `main':
/tmp/submission/20230922/07/02/hackerrank-51bcd6649c2f91f13fdb4d46c76e7fe6/code/Solution.cpp:44: undefined reference to `preparebiryani(int, bool&, char*, double*)'
collect2: error: ld returned 1 exit status
```

No Comments