**VISVESVARAYA TECHNOLOGICAL UNIVERSITY**

**“Jnana Sangama”, Belagavi-560018, Karnataka**



**DESIGN AND ANALYSIS OF ALGORITHMS LABORATORY**

**WITH MINI PROJECT-** **19IS4DLADA**

**REPORT**

**On**

**“COIN EXCHANGER”**

**BACHELOR OF ENGINEERING**

In

**INFORMATION SCIENCE AND ENGINEERING**

Submitted by

**Nisarga K(1DS20IS065)**

**Nithya M(1DS20IS067)**

**Saijyoti Meti(1DS20IS085)**

**Sanjana H Goud(1DS20IS092)**

**LAB IN-CHARGE**

Mrs. Radhika T V

Mrs. Shilpashree S

Mrs. Sreevidya B S



**2021-22**

**Department of Information Science and Engineering**

**DAYANANDA SAGAR COLLEGE OF ENGINEERING**

**SHAVIGE MALLESHWARA HILLS, KUMARASWAMY LAYOUT**

**Bengaluru-560078**

**DAYANANDA SAGAR COLLEGE OF ENGINEERING**

**Shavige Malleshwara Hills, Kumaraswamy Layout-560078**

**Department of Information Science and Engineering**



**2021-22**

**CERTIFICATE**

This is to certify that Design and Analysis of Algorithms Laboratory with Mini Project Work (19IS4DLADA) entitled **“COIN EXCHANGER”** is a bonafide work carried out by **Nisarga K(1DS20IS065) , Nithya M(1DS20IS067) , Saijyoti Meti(1DS20IS085) , Sanjana H Goud(1DS20IS092)** in the partial fulfillment for the 4th semester of **Bachelor of Engineering in Information Science and** **Engineering** of theVisvesvaraya Technological University, Belgaviduring the year 2021-22.

|  |  |  |
| --- | --- | --- |
|  |  | *Signature of Lab-In charge* |
|  |  |  |

Name of the Examiners Signature with Date

1.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **Sr. No** | **Topics** | **Page No.** |
|  | Introduction   * Definition * Example | 01 |
|  | System Requirements   * Hardware requirements * Software requirements | 02 |
|  | Algorithm   * Greedy algorithm * Tracing | 03  04 |
|  | Implementation |  |
|  | 1. Pseudocode | 05 |
|  | 1. Snapshots of Results | 06-08 |
|  | Time Complexity | 09 |

**1.INTRODUCTION**

**1.1 Definition:**

Coin exchange is finding the minimum number of coins that can give change for given amount of money. If have unlimited supply of coins of different country denominations then, the solution would be the minimum number of coins and/or notes needed to make the change.

The coin exchange problem can be implemented mainly in wending machines where human intervention will be less

**1.2 EXAMPLE**

Consider Indian Currency Denominations:

2000,500,200,100,50,20,10,5,2,1

How would you give change with coins of these denominations of, say, 63 Rs

The answer would be 50Rs note, 10Rs note,2-rupee coin and 1rupee coin.

**2.SYSTEM REQUIREMENTS**

**2.1 Hardware requirements:**

RAM Size used: 8GB

Min Hard Drive Space used: 25GB

Processor Used: Intel iRIS

**2.2 Software requirements:**

OS Required: Microsoft Windows 10

OS used: Windows 10

Programming Language: C++

Software Used: Visual Studio Code

**3. ALGORITHM**

* 1. **Greedy Algorithm**

The approach applied in the change-making problem is called **Greedy** **approach**. A common approach would be to take coins with greater value first. This can reduce the total number of coins needed. Start from the largest possible denomination and keep adding denominations while the remaining value is greater than 0.

For successful Greedy approach it should satisfy following three conditions:

1. Feasible- it has to satisfy the problem's constraints.
2. Locally optimal- it has to be the best local choice among all feasible choices available on that step.
3. Irrevocable- once made, it cannot be changed on subsequent steps of the algorithm.

**Algorithm:**

1. Sort the array of coins in decreasing order

2. Initialize result set as empty

3. Find the largest denomination that is smaller than entered amount.

4. Add found denomination to result. Subtract value of found denomination from amount.

5. If amount becomes 0, then print result.

6. Else repeat steps 3 and 4 for new value of amount.

**3.2 Tracing**

Input:

Value=147 Result set: empty

Check denominations:1,2,5,10,20,50,100,200,

500,2000

100 <=147

So, push 100 in result set.

New Value= 147-100=47

Value= 47 Result set: {100}

Check denominations:1,2,5,10,20,50

So, push 20 in result set.

New Value=47-20=27

Value=27 Result set: {100,20}

Check denominations: 1,2,5,10,20

20<=27

push 20 in to result set.

New value=27-20=7

Value=7 Result set: {100,20,20}

Check denominations:1,2,5

5<=7

So, push 5 in to result set

New value =7-5=2

Value=2 Result set={100,20,20,5}

Check denominations:1,2

2<=2

Push 2 in to result set

New value =2-2=0

Value=0 Since, No denominations less than 0

We print the result set.

**4. IMPLIMENTATION**

**4.1 Pseudocode**

Void countCurrencyINR (int amount)

{

int noteCounter[10];

int INR [10] = {2000, 500, 200, 100,

50, 20, 10, 5, 2,1};

for (int i = 0; i < 10; i++) {

if (amount >= INR[i]) {

noteCounter[i] = amount / INR[i];

amount = amount - noteCounter[i] \* INR[i];

}

}

cout << "Currency Count in INR" << endl;

for (int i = 0; i < 10; i++) {

if (noteCounter[i]! = 0) {

cout<<"Rs " << INR[i] << “: “

<< noteCounter[i] << endl;

}

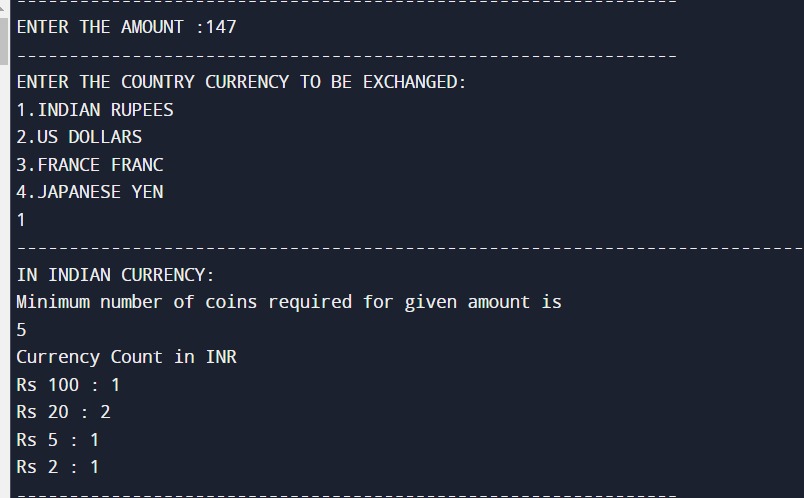
}

}

* 1. **Snapshots of Result**



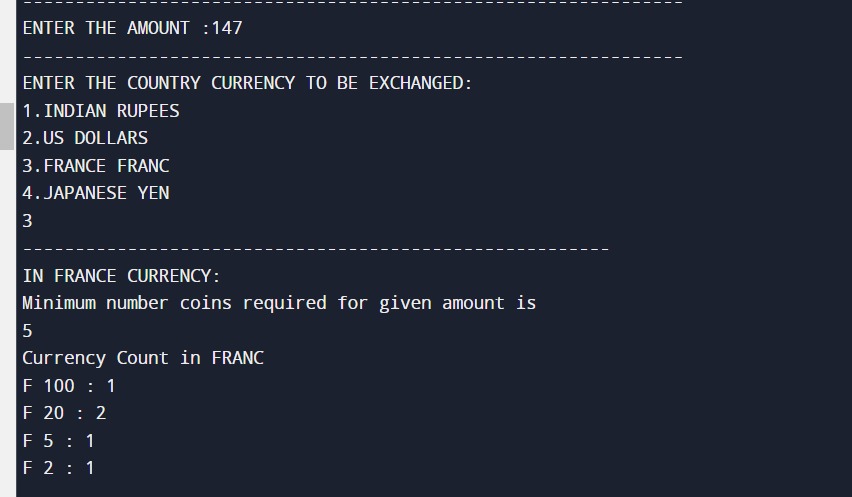
**Fig 1. Entering Amount (Input)**

****

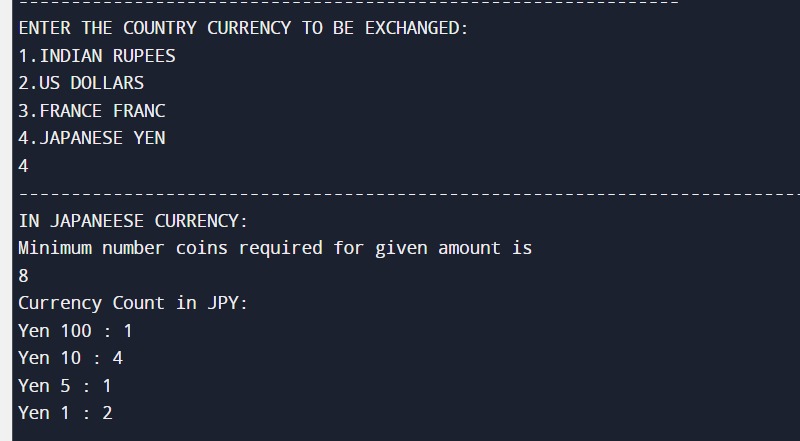
**Fig 2 Output for Indian Currency**



**Fig 3 Output for US Currency**

****

**Fig 4 Output for France Currency**



**Fig 5 Output for Japan Currency**

**5.TIME COMPLEXITY**

**Time Complexity:**

Time complexity of greedy coin change will be:

* For sorting n coins **O(nlogn).**
* For loop worst case is **O(m).**

Where m-number of denominations.

Worst case: If all we have is the coin with 1-denomination.

If amount(n) is smaller than highest available denominations(m) then for loop will run many times which will lead to more time complexity.

**Space complexity: O(m)**