****



Mini Project Report

On

**“Quests”**

Under the guidance of

**Mrs. Mrunal Deshpande**

**School of Computer Science & Engineering**

**Academic year: 2021 - 22**

**CERTIFICATE**

This is to certify that the mini project report entitled

**“Quests”**

submitted by

**Yogesh Ahuja(24)**

**Aaditya Gopalani(22)**

**Shivani Patil(26)**

**Aditya Patil(30)**

In the partial fulfilment for the Semester IV of S. Y. B. Tech of Computer Science & Engineering is a record of work carried out by the students mentioned above under the guidance and supervision of Mrs. Mrunal Deshpande during the academic year 2021-22.

**Place: SGU, Atigre**

**Date:**

**Mrs. Mrunal Deshpande Dr. B. Suresh Kumar**

**Mini Project Guide Head-SOCSE**

**ACKNOWLEDGEMENT**

We express our sincere thanks to **Mrs. Mrunal Deshpande** and **Dr. B. Suresh Kumar, Head-SOCSE** whose supervision, inspiration and valuable guidance , helped us a lot to complete our mini project work.

Their guidance proved to be the most valuable to overcome all the hurdles in the fulfilment of this mini project work.

Last but not the least, this acknowledgement would be incomplete without rendering our sincere gratitude to all those who have helped us in the completion of mini project work.

Sincerely,

1. Yogesh Ahuja(24)

1. Aaditya Gopalani(22)

1. Shivani Patil(26)

1. Aditya Patil(30)

**INDEX**

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Chapter** | **Page No.** |
| 1 | Introduction | 1 |
| 2 | Problem Statement | 2 |
| 3 | Literature Review | 3 |
| 4 | Objectives | 4 |
| 5 | Methodology | 5-7 |
|  | 5.1 General Method | 5 |
|  | 5.2 Algorithm | 5 |
|  | 5.3 Flowchart | 6-7 |
| 6 | System Requirements | 8 |
| 7 | Results | 9-10 |
| 8 | Conclusion | 11 |
| 9 | References | 12 |

1. **Introduction**

We have to play a computer game called as Quest.

In this game the player will enter number of quest he/she want to complete, number of XP’s for each level and XP multiplier.

After that player will give input for quest i.e. number of xp of points and difficulty level for each quest.

After that the program will calculate the maximum number of XP’s that a player can earn for the input given by the player

(XP = Experience Points)

For example in the sample input, the maximum XPs you can earn is 43, which is done as follows. First complete the second quest (you earn 4 XPs, because you are at level 0, and you completed a quest with target difficulty level 2). Then complete the first quest (you earn 30 XPs, because you are still at level 0, and the target difficulty level is 1). With 34 XPs, you are now level 3. Finally, complete the third quest (for which you earn 9 XPs, without the multiplier, since you are already at level 3).

1. **Problem Statement**

To relax before competing in the ICPC World Finals, you have decided to play a computer game called Quests. You have played it a number of times already, and now you want to achieve a perfect playthrough—to prepare for your perfect playthrough of the finals!

In the game, you have to complete a number of quests, and you earn experience points (XPs) for completing each one. The total number of XPs you have earned at any time determines your current level. You reach a new level for every v XPs that you earn. Formally, your level at any time is the largest integer L such that you have at least L · v XPs.

Each quest is assigned a number x of XPs and a target difficulty level d. If you complete the quest when your level is at least d, you earn x XPs. However, if you complete the quest when your level is less than d, you will earn c · x XPs. The constant c is an XP multiplier that results in a bonus for completing a quest when you are at a lower level than the recommended level d.

You know all the n quests and their respective x and d numbers by heart (and you know the numbers v and c as well—you have played this game a lot). You are also skilled enough to complete any quest, regardless of its target difficulty level and your level. You want to complete all the quests in an order that will allow you to earn the largest possible number of XPs.

1. **Literature Review**
2. **Sorting Algorithms:-**

<https://www.geeksforgeeks.org/sorting-algorithms/>

Bubble Sort is the simplest sorting algorithm that works by repeatedly swapping the adjacent elements if they are in the wrong order. Function always runs O(n^2) time even if the array is sorted.

The key process in quickSort is partition(). Target of partitions is, given an array and an element x of array as pivot, put x at its correct position in sorted array and put all smaller elements (smaller than x) before x, and put all greater elements (greater than x) after x. All this should be done in linear time. Time taken by QuickSort, in general, can be written as following.

T(n) = T(k) + T(n-k-1) + \theta(n)

So, we will be using the quick sort.

1. **Quests:-**

<https://en.wikipedia.org/wiki/Quest>

A quest is a journey toward a specific mission or a goal. The word serves as a plot device in mythology and fiction: a difficult journey towards a goal, often symbolic or allegorical. Tales of quests figure prominently in the folklore of every nation[1] and ethnic culture. In literature, the object of a quest requires great exertion on the part of the hero, who must overcome many obstacles, typically including much travel.

1. **Experience Points:-**

<https://en.wikipedia.org/wiki/Experience_point>

An experience point (often abbreviated as exp or XP) is a unit of measurement used in some tabletop role-playing games (RPGs) and role-playing video games to quantify a player character's life experience and progression through the game. Experience points are generally awarded for the completion of missions, overcoming obstacles and opponents, and for successful role-playing.

1. **Objective**
   * The main objective of Quest game is to find maximum of XP’s(Experience Points) from the give input by the player.
   * To complete any quest, regardless of its target difficulty level.
   * To complete all the quests in an order that will allow you to earn the largest possible number of XP’s.
2. **Methodology**
   1. **General Method:-**
3. Modules:
4. Inputs – NumberOfQuests(n), ExperiencePoints(v), XpMultiplier(c):

The all the necessary inputs from the player-n, v and c.

* + 1. Deadline:

Sort all the inputs according to the deadline(xp\*XpMultiplier + DifficultyLevel\*ExperiencePoints) in ascending order.

* + 1. Maximum ExperiencePoints:

Complete the quest one by one and give the output i.e. Maximum number of Experience Points that a player can earn.

* 1. **Algorithm:-**

**Step 1:-** Create a structure for quest with properties-Experience points, target difficulty level and deadline.

**Step 2:**- Create array of structure quest.

**Step 3:**- Create a function for comparison.

**Step 4:**- Declare variables for CurrentLevel and MaximumXp and initialize to zero.

**Step 5:**- Take inputs for NumberOfQuests(n), ExperiencePoints(v), XpMultiplier(c).

**Step 6:**- According to the value of n take inputs for quest i.e. xp and target difficulty level of that quest.

**Step 7:**- Calculate deadline = (xp \* c) + (target \* v)

**Step 8:**- Sort the array of structure quest.

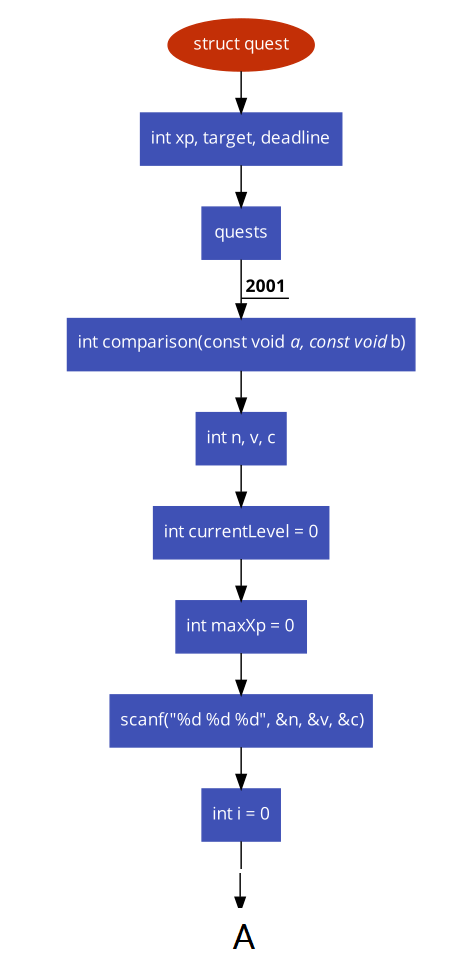
**Step 9:**- Create loop according to the value of n and complete the quest.

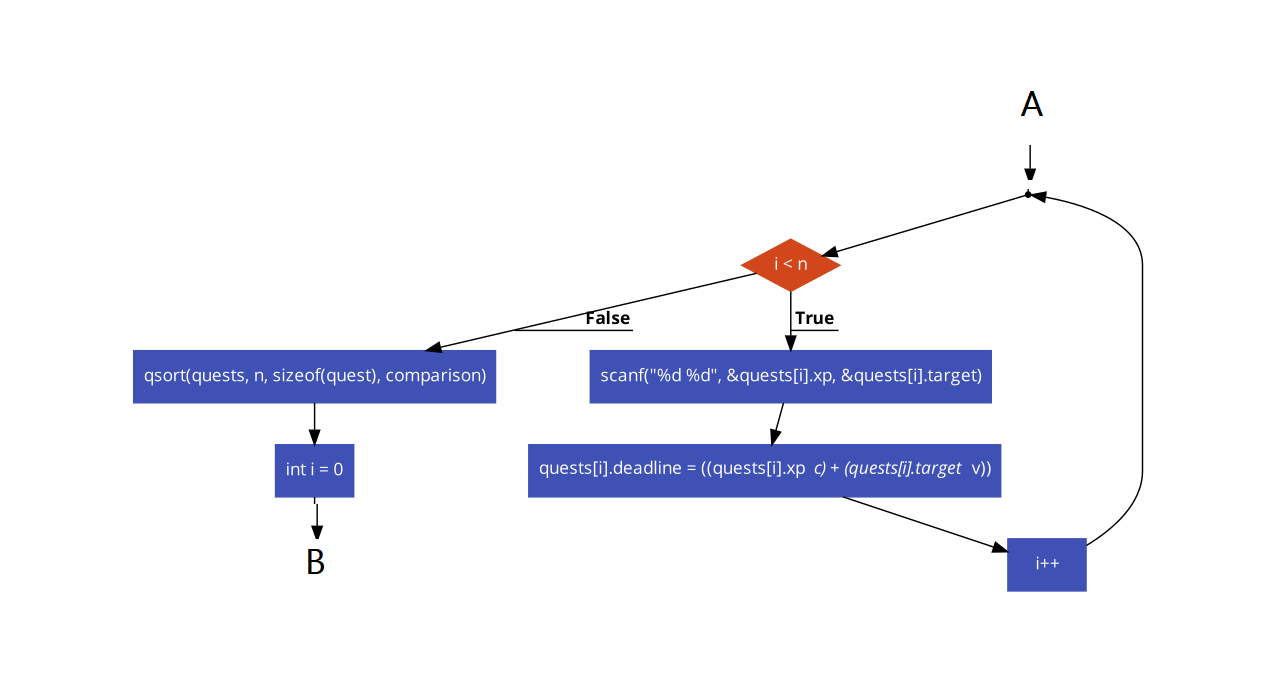
**Step 10:**- if CurrentLevel is less than target difficulty level of that quest then multiply xp of that quest with the xp multiplier(c) and add to the MaximumXp variable.

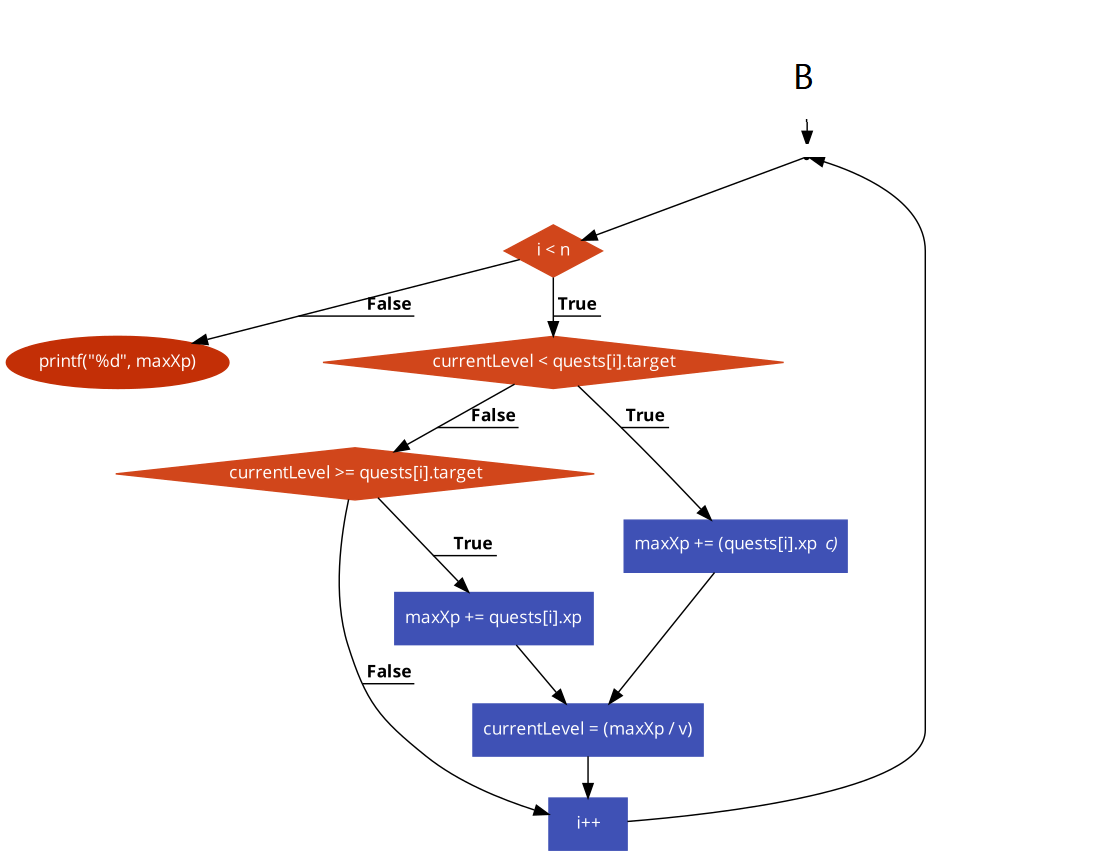
**Step 11:**- If CurrentlLevel is greater than or equal to target difficulty level of that quest then directly add xp to the MaximumXp variable.

**Step 12:**- Print the value of MaximumXp variable.

* 1. **Flowchart:-**

****

****

****

1. **System Requirements**

* **Software requirements:**
  + - C compiler
* **Hardware requirements:**
* Processor Intel i3/i5/i7
* RAM minimum 4 GB
* Hard disk minimum 15 GB
* internet connection

1. **Results**

#include<stdio.h>

#include<conio.h>

#include<stdlib.h>

#define MAXN 2001

struct quest

{

    int xp, target, deadline;

}quests[MAXN];

typedef struct quest quest;

int comparison(const void \*a, const void \*b)

{

    (void)a; (void)b;

    return rand() % 2 ? +1 : -1;

}

int n, v, c;

int main()

{

    int total = 0;

    int currentLevel = 0;

    int maxXp = 0;

    scanf("%d %d %d", &n, &v, &c);

    for(int i = 0; i < n; i++)

    {

        scanf("%d %d", &quests[i].xp, &quests[i].target);

        quests[i].deadline = ((quests[i].xp \* c) + (quests[i].target \* v));

        total += quests[i].xp;

    }

    qsort(quests, n, sizeof(quest), comparison);

    for(int i = 0; i < n; i++)

    {

        if(currentLevel < quests[i].target)

        {

            maxXp += (quests[i].xp \* c);

            currentLevel = (maxXp / v);

        }

        else if(currentLevel >= quests[i].target)

        {

            maxXp += quests[i].xp;

            currentLevel = (maxXp / v);

        }

    }

    printf("%d", maxXp);

}

**Inputs and Outputs:-**

|  |  |
| --- | --- |
| Sample Input 1 | Sample Output 1 |
| 3 10 2 | **43** |
| 15 1 |  |
| 2 2 |  |
| 9 1 |  |

|  |  |
| --- | --- |
| Sample Input 2 | Sample Output 2 |
| 2 20 3 | **45** |
| 6 1 |  |
| 9 2 |  |

|  |  |
| --- | --- |
| Sample Input 3 | Sample Output 3 |
| 4 15 2 | **160** |
| 10 1 |  |
| 20 2 |  |
| 30 3 |  |
| 40 4 |  |

1. **Conclusion**

We studied about the computer game called Quest. In this game, player gave theinputs for xp’s, xp multiplier and number of quest. And after that we first calculated the deadline and after calculating we started completing the quest and found the maximum number of experience points that the player can earn for the inputs he/she has given to the program.

1. **References**
2. [**https://icpc.global/worldfinals/problems/2020+ACM-ICPC+World+Finals/icpc2020.pdf**](https://icpc.global/worldfinals/problems/2020+ACM-ICPC+World+Finals/icpc2020.pdf)
3. [**https://www.geeksforgeeks.org/sorting-algorithms/**](https://www.geeksforgeeks.org/sorting-algorithms/)
4. [**https://en.wikipedia.org/wiki/Quest**](https://en.wikipedia.org/wiki/Quest)
5. [**https://en.wikipedia.org/wiki/Experience\_point**](https://en.wikipedia.org/wiki/Experience_point)