Assignment 8: Heavy Numbers

EC602 Design by Software

Fall 2021

Contents

1 Introduction

1.1 Assignment Goals

The assignment goals are to help you learn about

- arithmetic operators
- number systems and bases
- sequences

1.2 Group Size

The maximum group size is 2 students.

1.3 Due Date

The assignment is due 2021-11-22 at 23:59:59

1.4 Assignment Value

This assignment is worth 5 points.

1.5 Late policy

Late assignments will be accepted until the beginning of the lecture immediately following the due date, or for 48 hours, whichever is less.

If the *natural grade* on the assignment is g, the number of hours late is h, and the number of hours between the assignment due time and the next class is H, the new grade will be

```
(h > H) ? 0 : g * (1- h/(2*H))
```

If the same assignment is submitted ontime *and* late, the grade for that component will be the maximum of the ontime submission grade and the scaled late submission grade.

1.6 Submission Link

You can submit here: heavy hw8 submit link

2 Heavy Numbers

2.1 Background on heavy numbers

2.1.1 The heavy sequence

A sequence of numbers (the heavy sequence)

$$y_0 y_1 y_2 y_3 \cdots y_n \cdots$$

is defined such that each number is the sum of digits squared of the previous number, in a particular base.

Consider numbers in base 10, with $y_0 = 12$

The next number in the sequence is $y_1 = 1^2 + 2^2 = 5$

The next number in the sequence is $y_2 = 5^2 = 25$

The next number in the sequence is $y_3 = 2^2 + 5^2 = 29$

2.1.2 Heaviness

It turns out that for each number y_0 and base N, the heavy sequence either converges to 1, or it does not.

A number whose sequence converges to 1 in base N is said to be "heavy in base N"

2.2 Program requirements

Write a program 'heavy.cpp' that reads a number y and a base N from the command line (both of which are written in base-10 notation) and returns whether that number y is heavy in the base N provided.

The return code of this program should be 1 if the number is heavy, and 0 if the number is not heavy.

Here are examples:

```
> heavy 4 10
> echo $?
0

> heavy 2211 10
> echo $?
1

> heavy 23 2
> echo $?
1

> heavy 10111 2
> echo $?
1

> heavy 12312 4000
> echo $?
0
```

2.2.1 Value Ranges

The number y will always be storable as a normal `int`, and the base N will also be storable as a normal `int`.

The number y will always be non-negative, and the base N will always satisfy $2 \leq N \leq 4000$

2.2.2 Restrictions

You may only include the libraries `vector` and `string`. No other includes are permitted.

Brackets are not allowed except when converting `argv`.

2.3 Checker

There is a checker available here:

hw8_heavy_check.py