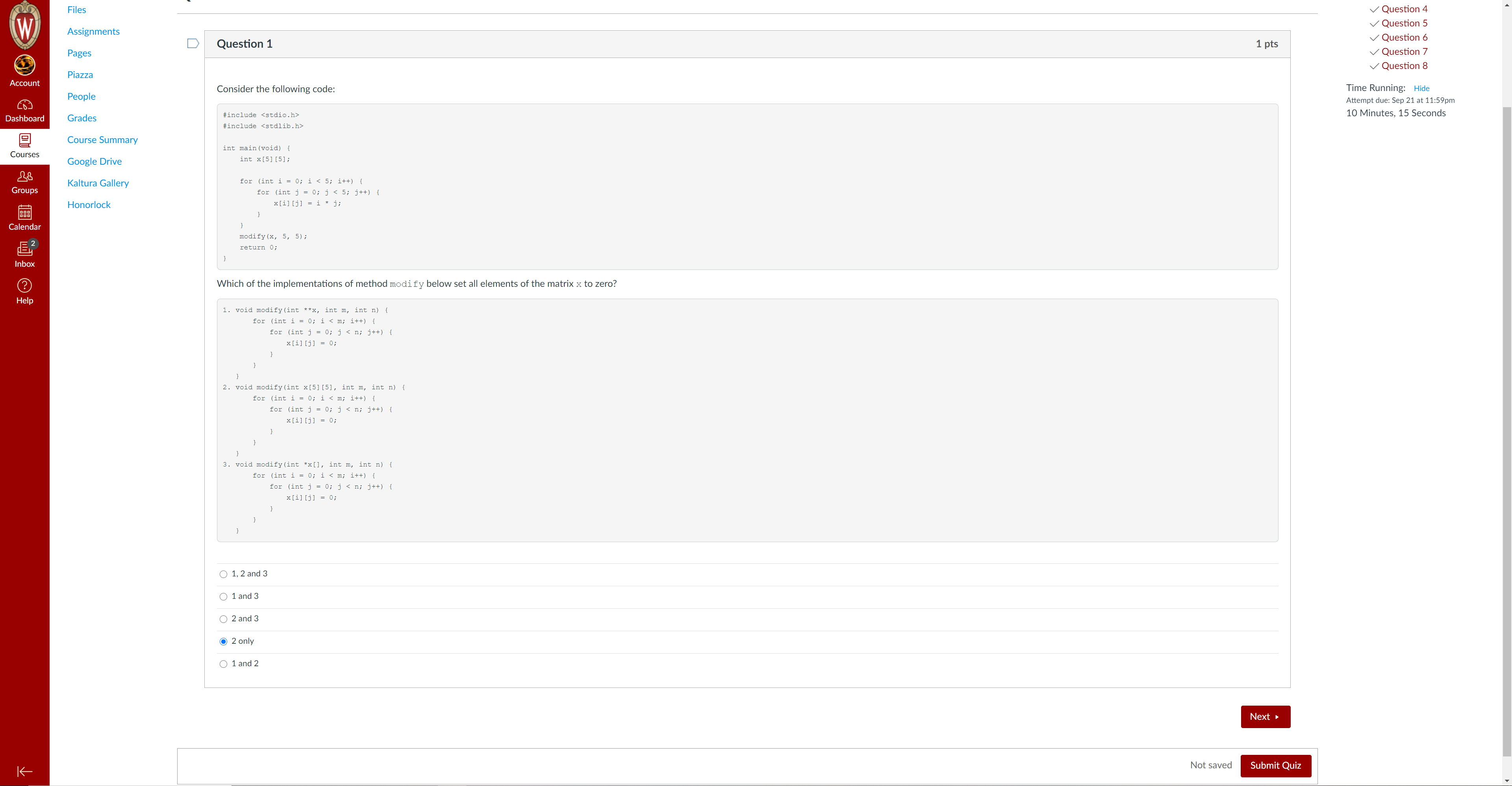
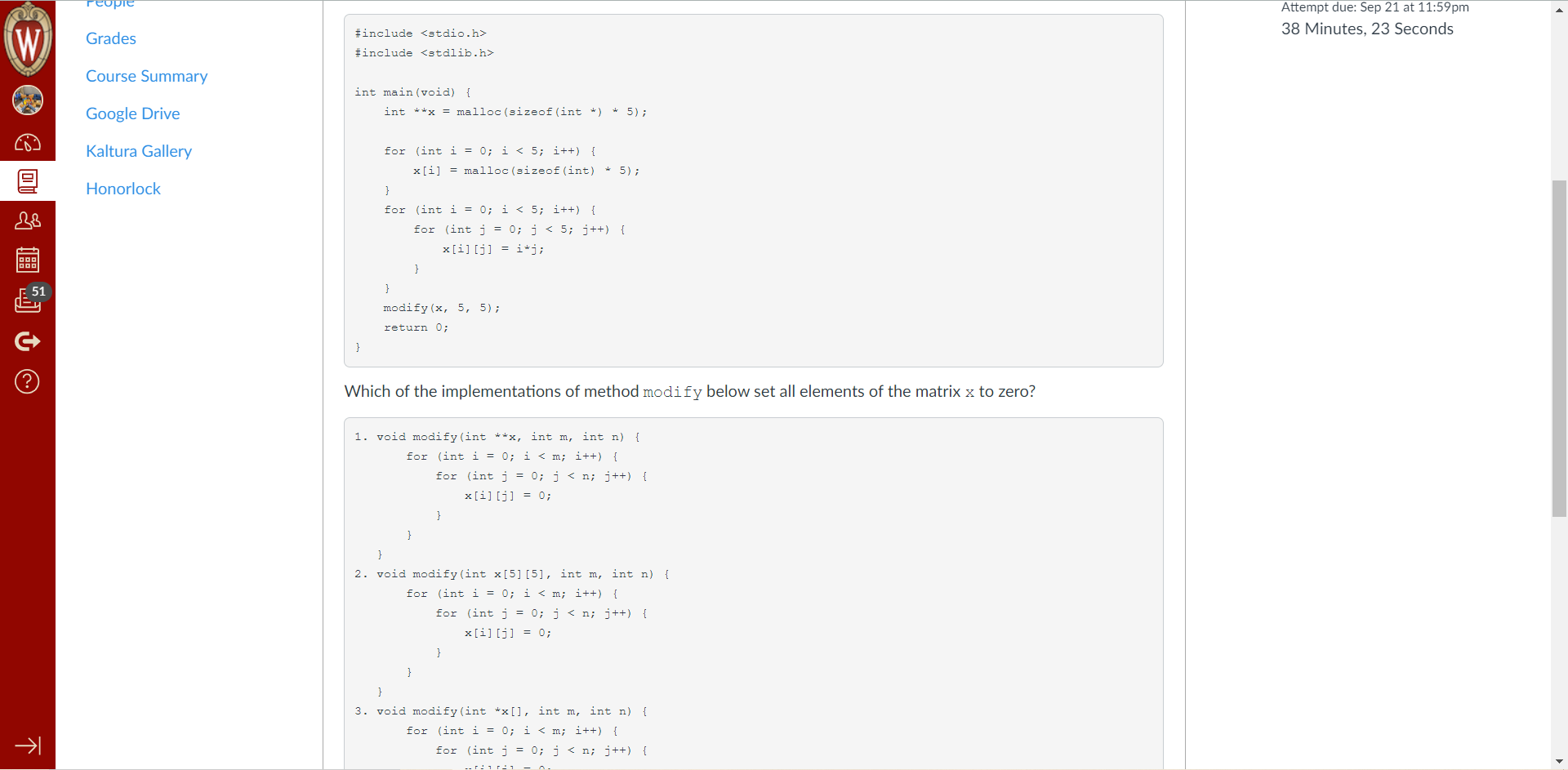
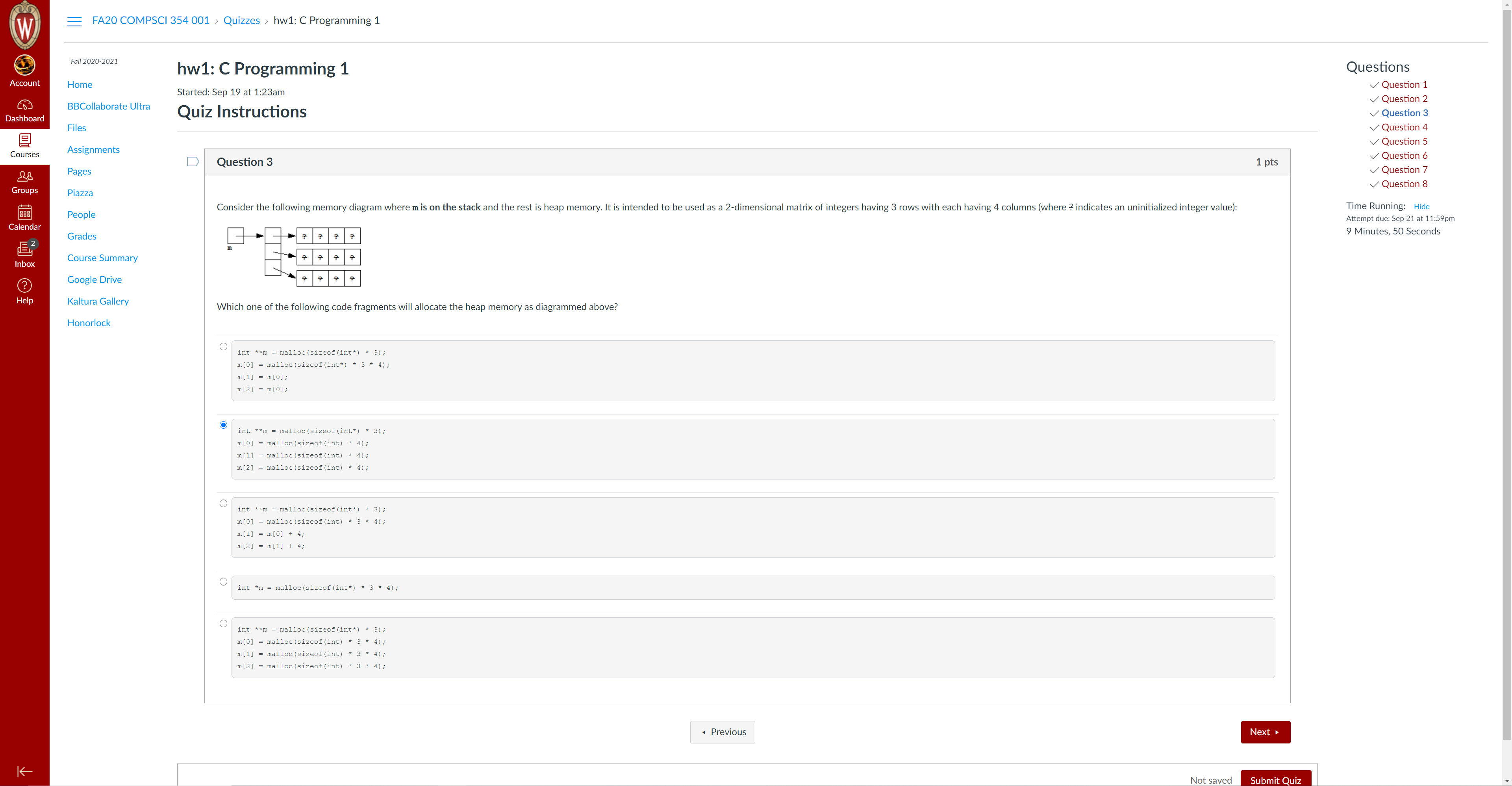
HW1

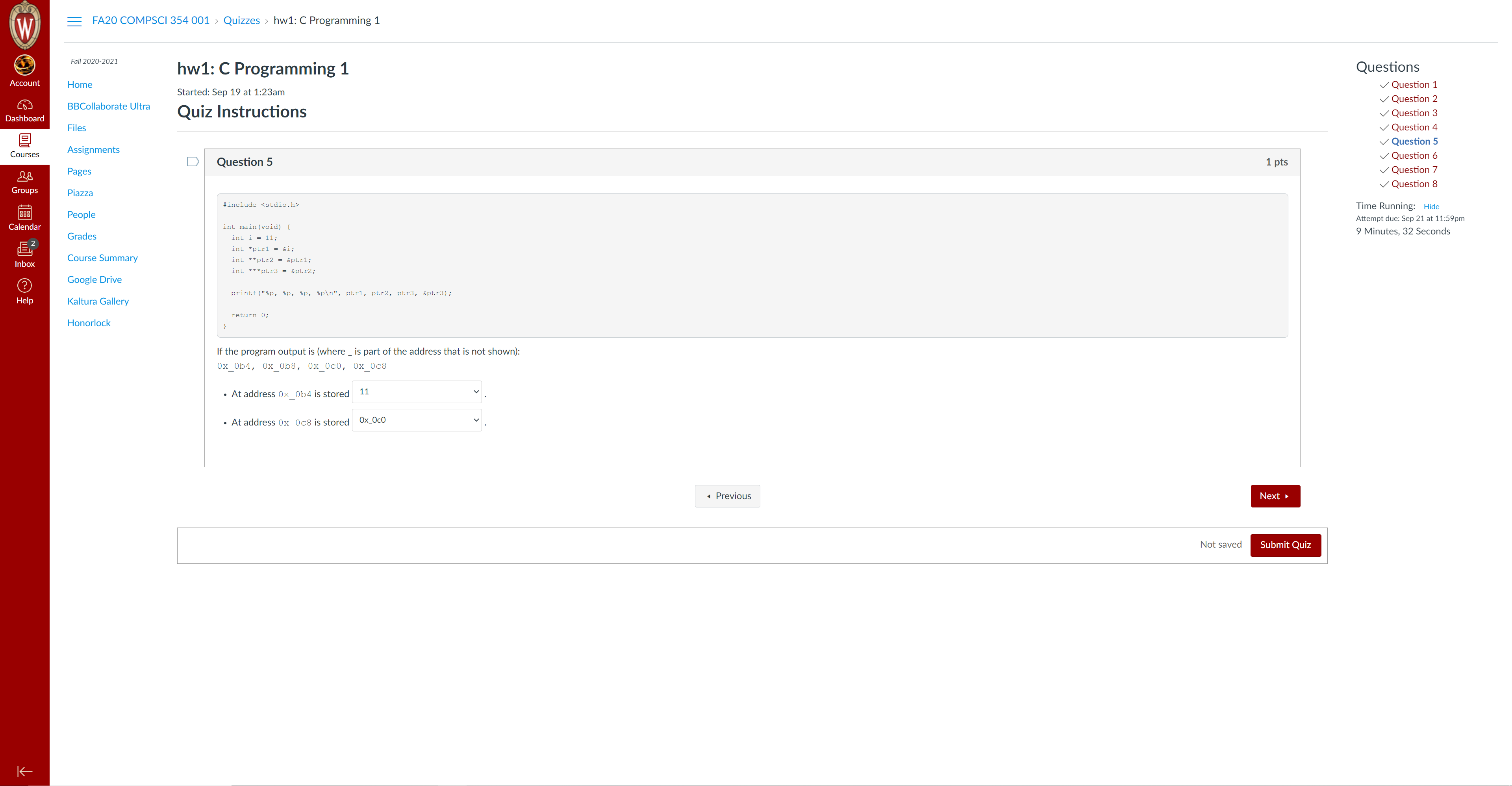


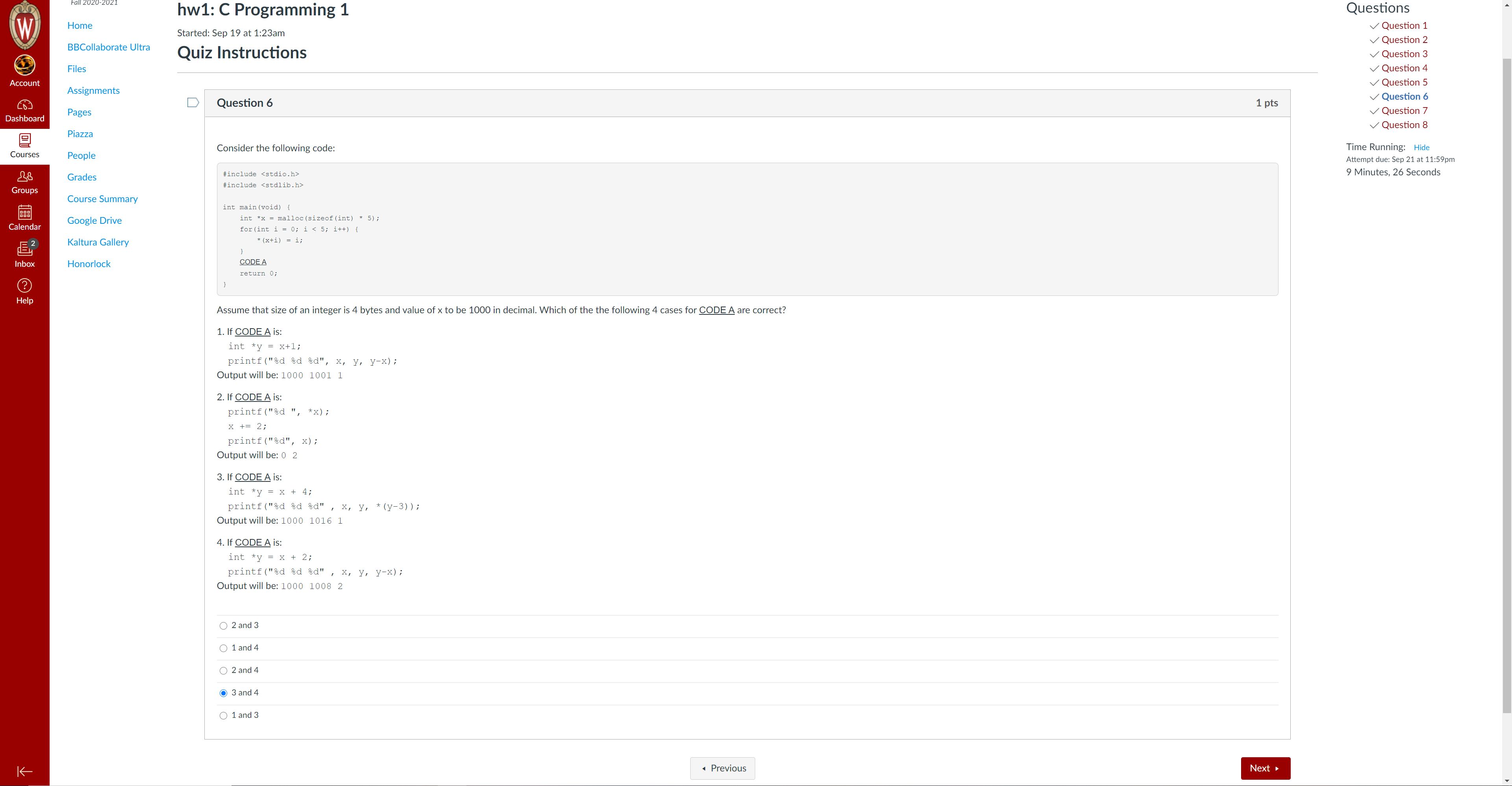
下面答案13

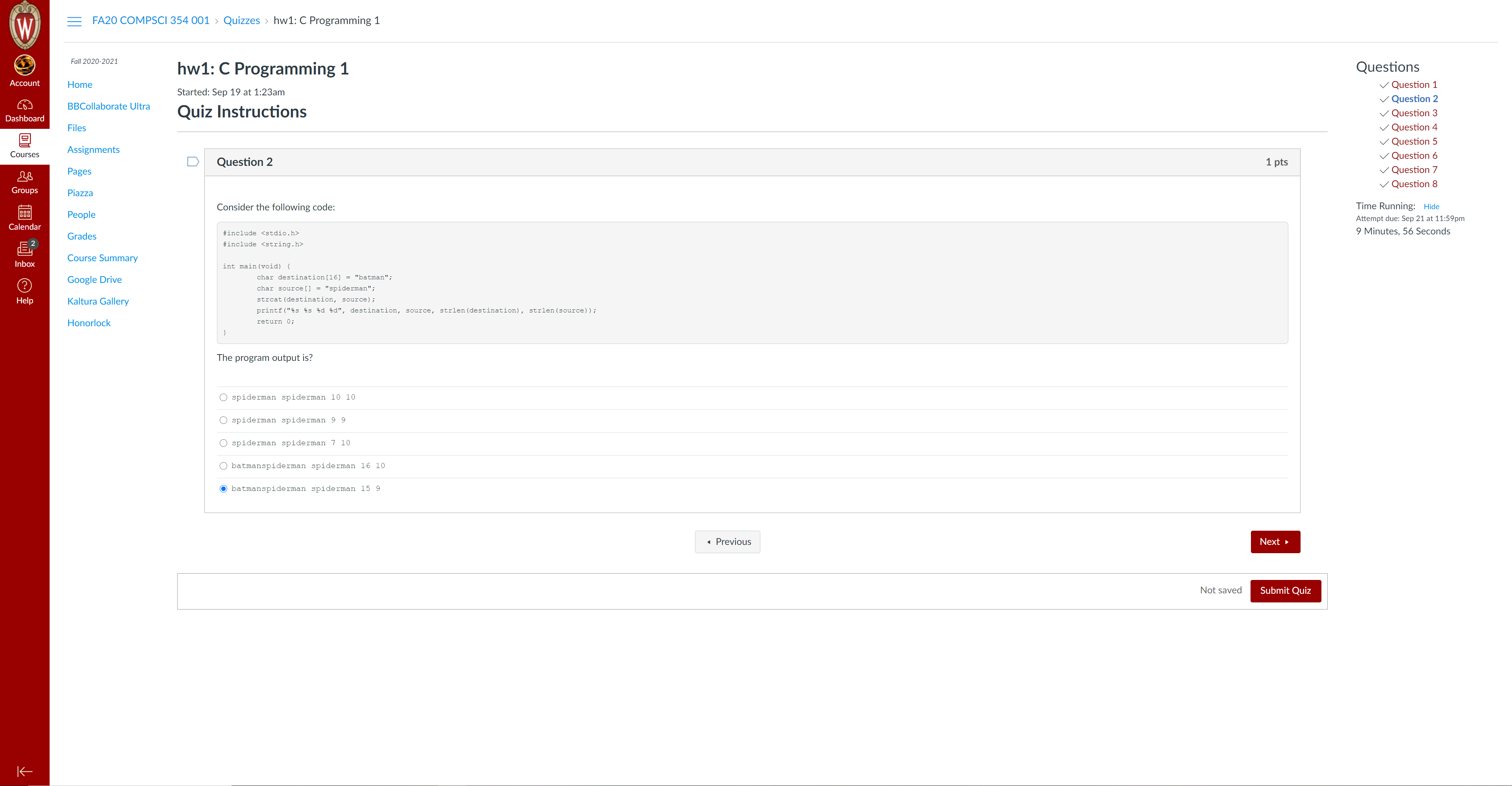


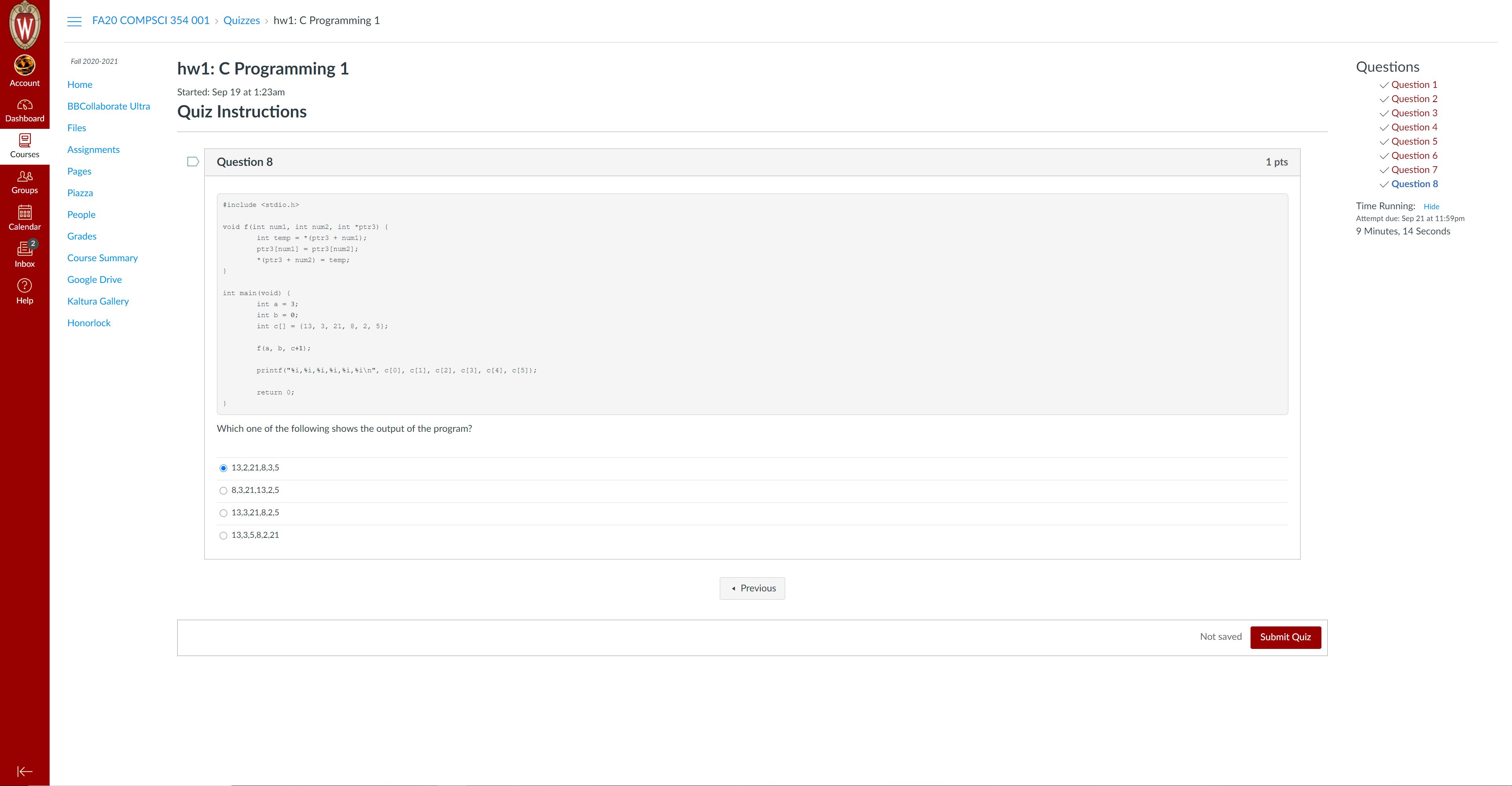


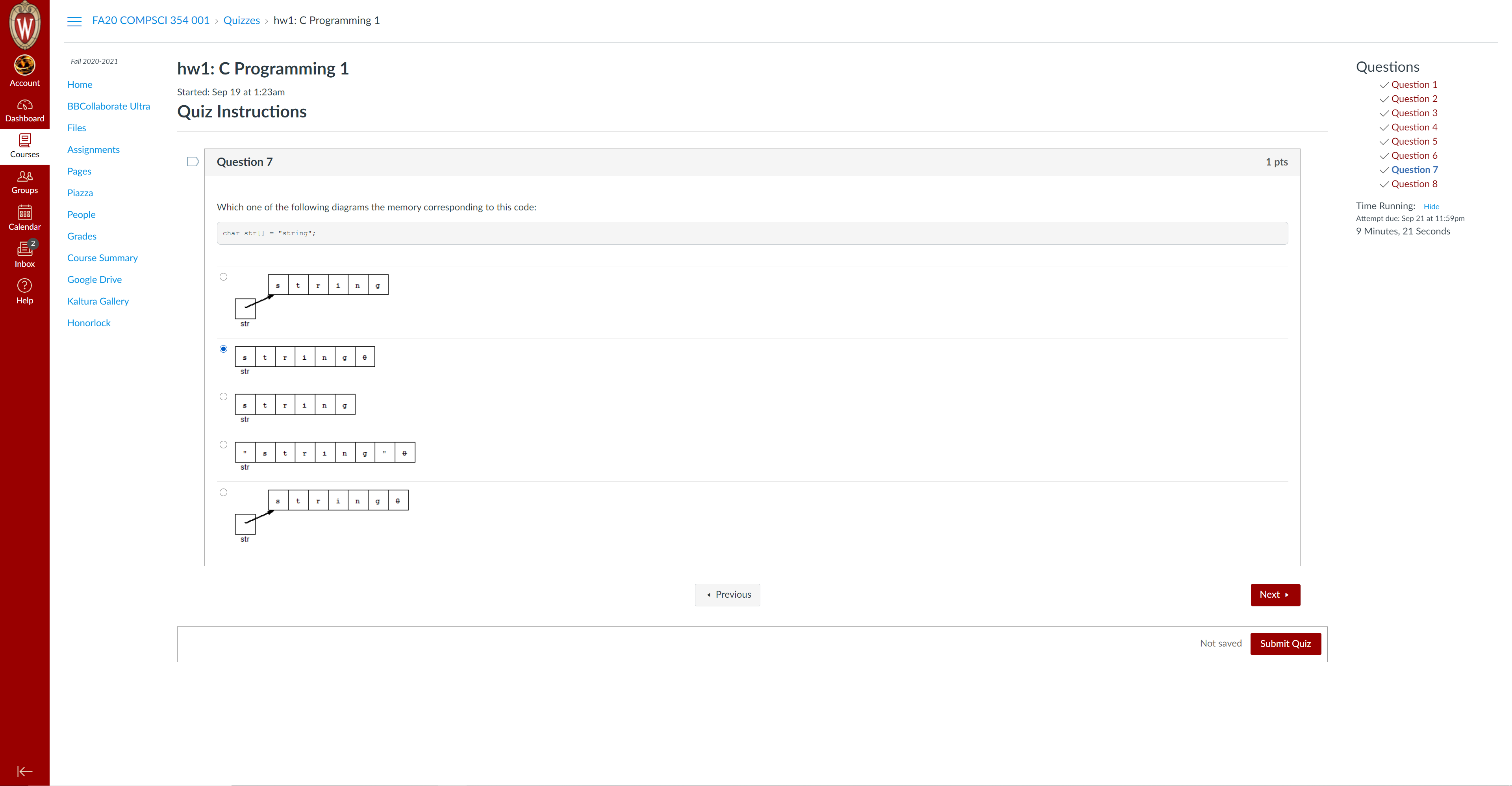




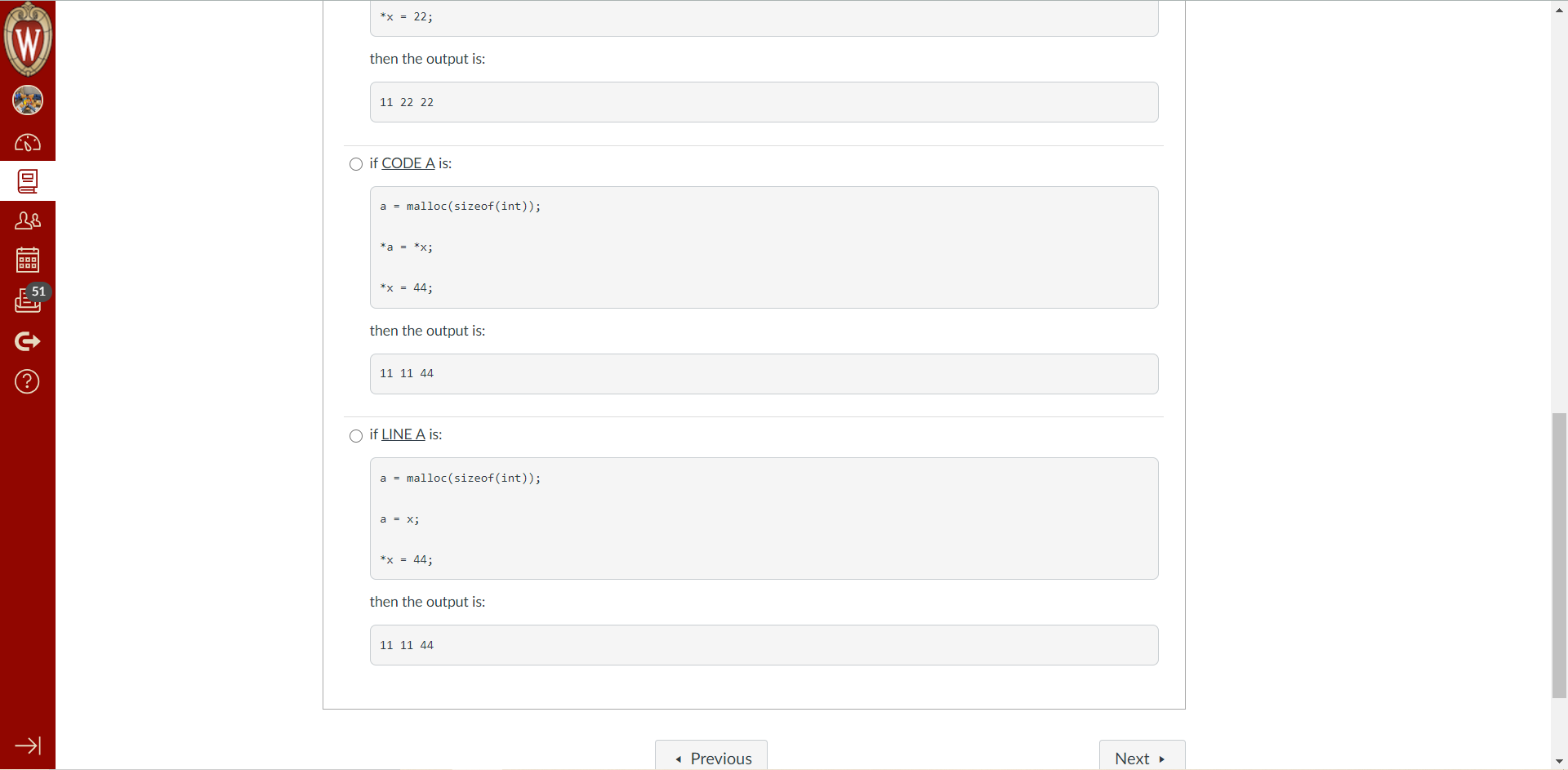
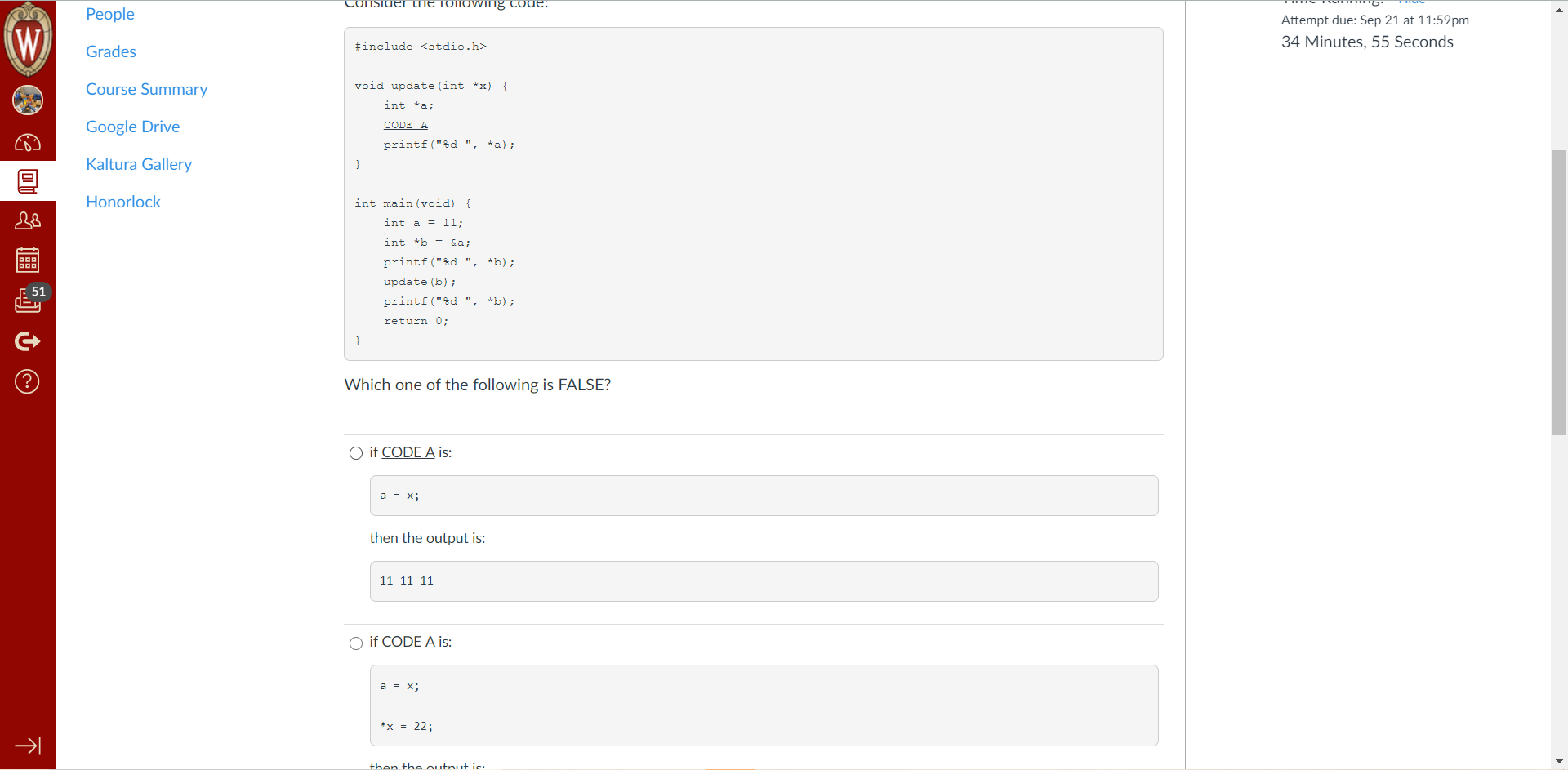


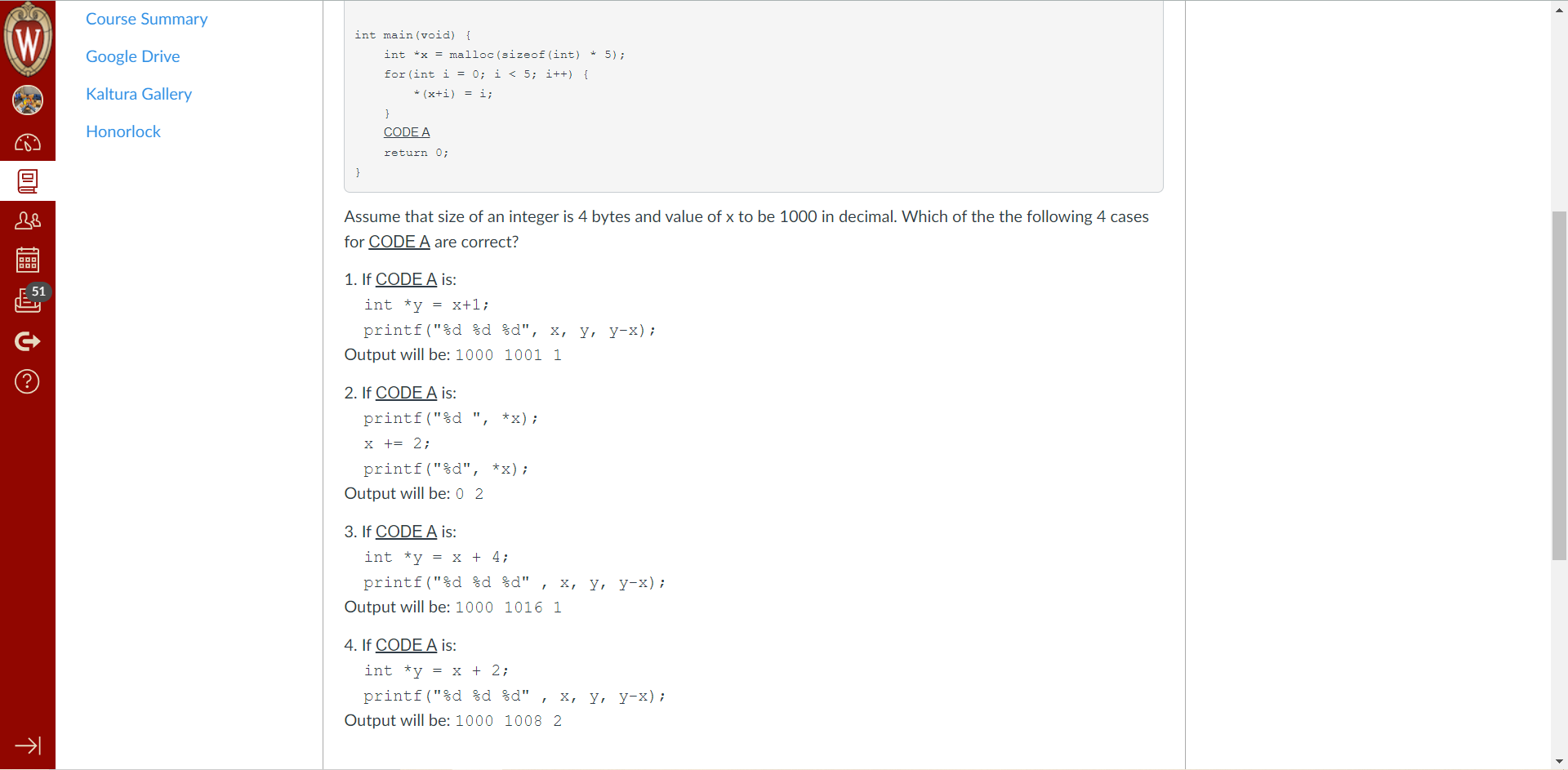
.

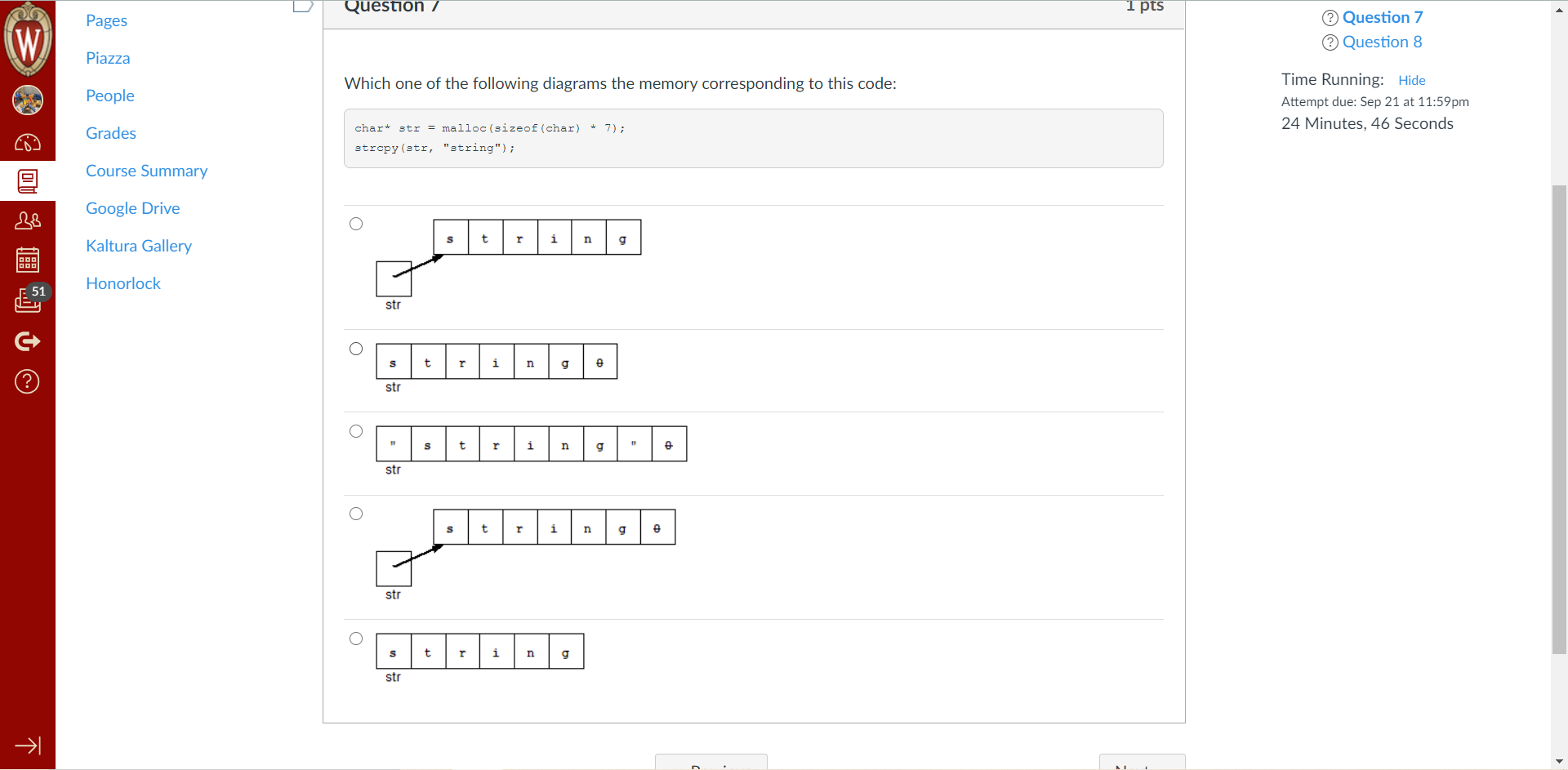
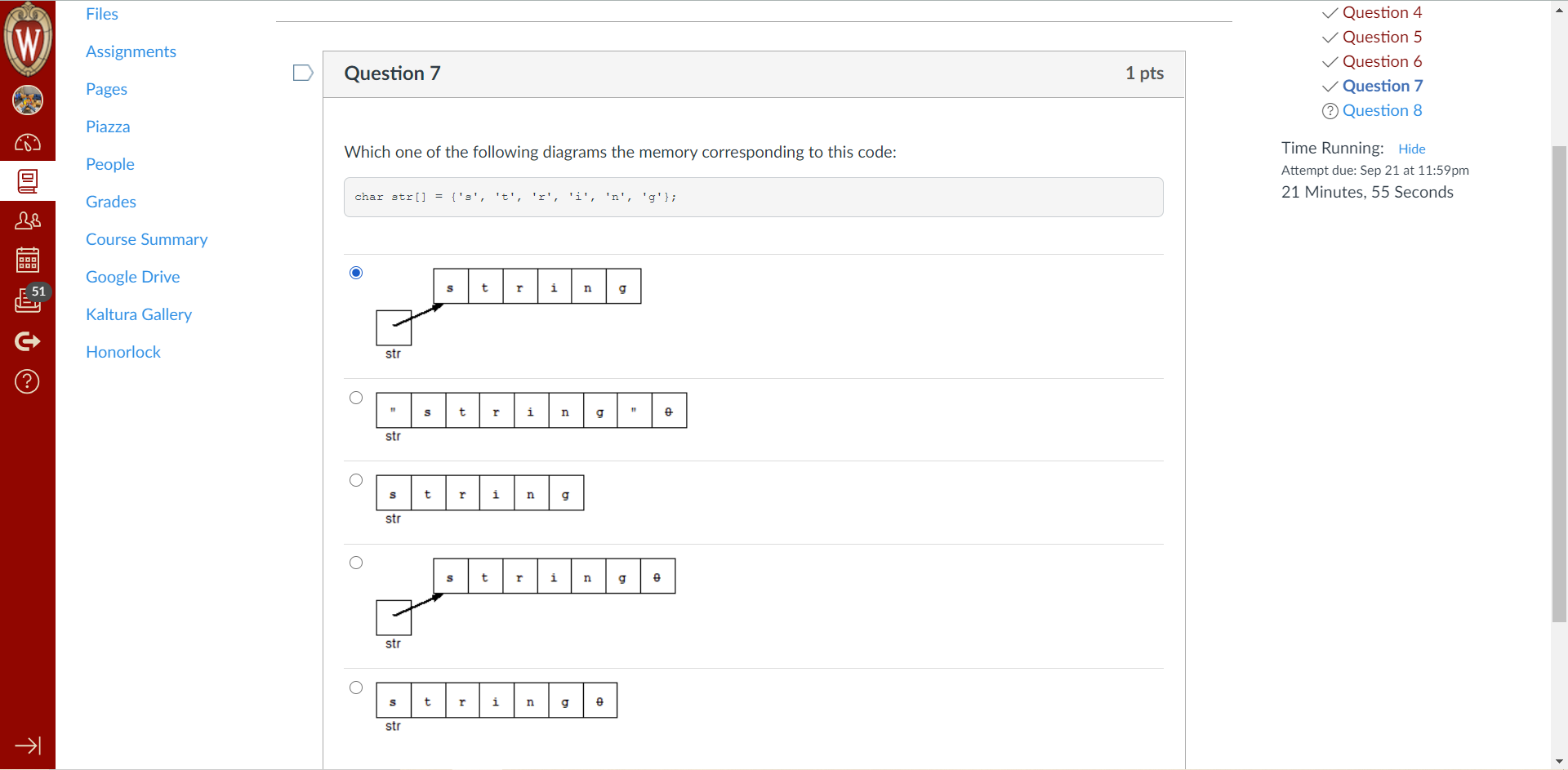




选B

最后一个

答案 2 4

DC

D

HW2

**Question 1 1 pts**

What is the output of the following program?

#include <stdio.h> int count = 10;

void func(int count) { printf("%d ", count);

count++;

}

int main(void) {

func(count); count++;

func(count); printf("\n"); return 0;

}

Group of answer choices 11 13

10 12

11 12

11 12

10 11

**Question 2 1 pts**

To avoid a limitation in Canvas, array code below has an extra space before the index.

typedef struct {

char name[ 11]; char \*type; float weight;

} Pokemon;

int main(void) {

Pokemon pokedex[ 7];

The description of pokedex is

"an array of Pokemon structures",

The description of pokedex[ 3] is

["a pointer to an array of Pokemon structures", "a Pokemon structure",

"a pointer to a Pokemon structure",

"an array of Pokemon structures", "an array of pointers to Pokemon structures"]

**Question 31 pts**

typedef struct {

} Team;

Teammate roster[33];

name[22];

char

size;

int

typedef struct {

} Teammate;

char position[22];

char lastname[22];

char firstname[22];

Assume team has been initialized as follows:

Team team;

Which of the following code fragments will access the lastname data member of the teammate at index 0. Select all that are correct.

Group of answer choices

(\*team->roster).lastname team.roster[0].lastname team->roster[0].lastname

(\*(\*team).roster).lastname

(\*team.roster).lastname

I'm a memory segment that a process uses whose memory allocations are automatic but not all are initialized. Who could I be?

Group of answer choices Heap

Data Segment Kernel

Stack

Code Segment

**Question 51 pts**

#include <stdlib.h> #include <string.h>

typedef struct { char title[55]; char \*author; int pages;

} Book;

typedef struct { int numBooks;

Book \*books[22];

} Shelf;

int main(void) {

Shelf bookShelf[11]; Book book;

bookShelf[3].books[7] = &book;

//statement(s) added here strcpy(bookShelf[3].books[7]->author, "Seuss");

Which of the following statements are required to allocate heap memory so that the last statement will make "Seuss" the author of a book in the bookshelf? Select all that are required and don't worry about their order if more that one statement is selected.

Group of answer choices

只选第四个

bookShelf[3].books[7] = malloc(sizeof(Book)); bookShelf[3].books[7]->author = malloc(sizeof(char)); bookShelf[3] = malloc(sizeof(Shelf)); bookShelf[3].books[7]->author = malloc(sizeof(char) \* 50);

bookShelf = malloc(sizeof(Shelf) \* 11);

**Question 61 pts**

Consider the following code:

#include <stdio.h> #include <stdlib.h> int a = 32;

int b;

int func(int arg) { static int tmp = 0; tmp++;

return tmp;

}

int main(int argc, char \*argv[]) { int \*p = malloc(sizeof(int));

\*p = 43;

char \*str = "where am I?";

printf("%s\n", str);

return 0;

}

Where are str and the string literal "where am I?" stored in the program's virtual address space?

Group of answer choices Code, Code

Stack, Stack Stack, Code 不对

Heap, Heap Stack, Heap应该是最后一个

Below is a basic implementation of the Linux command "cat". This command is used to print the contents of a file on the console/terminal window.

#include <stdio.h> #include <stdlib.h>

int main(int argc, char\* argv[]) {

FILE \*fp;

if(2 != argc) {

printf("Usage: cat <filename>\n"); exit(1);

}

if ((fp = fopen(argv[1], "r")) == NULL) {

fprintf(stderr, "Can't open input file %s\n", ar

gv[1]);

exit(1);

}

char buffer[256];

while (fgets(**X**, 256, fp) != NULL) fprintf(**Y**, "%s", buffer);

fclose(**Z**); return 0;

}

Which one of the following replacements for **X**, **Y** and **Z** will result in correct execution?

Group of answer choices

X = buffer, Y = stdin, Z = fp

X = buffer, Y = stdout, Z = fp 应该是这个

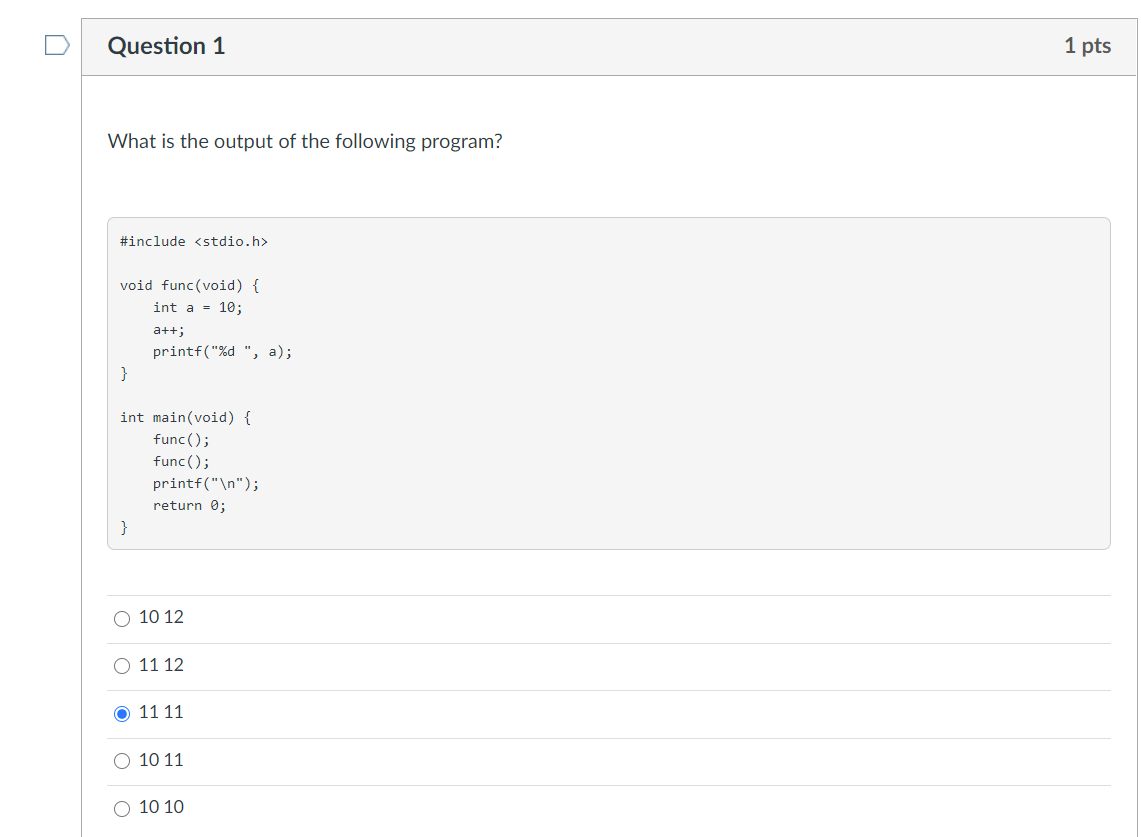
X = buffer, Y = fp, Z = fp 不对

X = fp, Y = stdout, Z = buffer X = buffer, Y = fp, Z = stdout

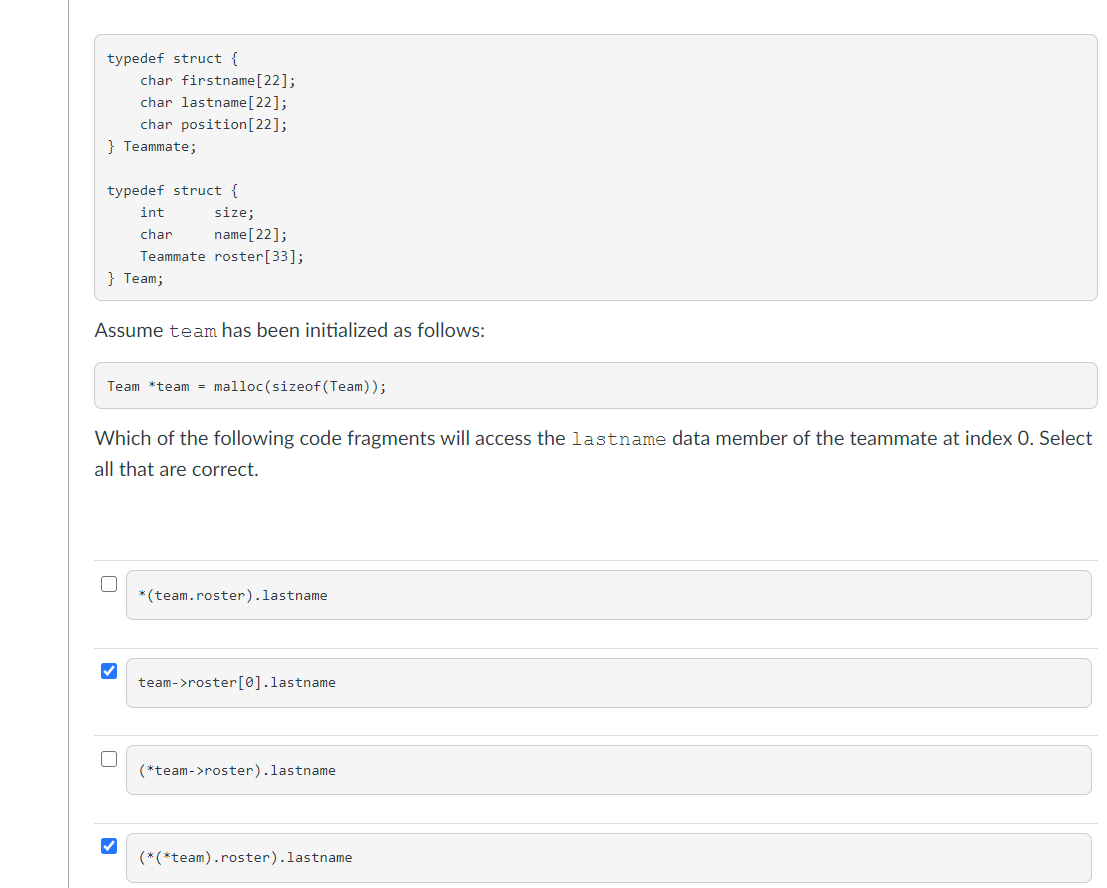
**Question 81 pts**

A process's memory segments in the virtual address space for a Linux based IA-32 system listed from low to high address are?

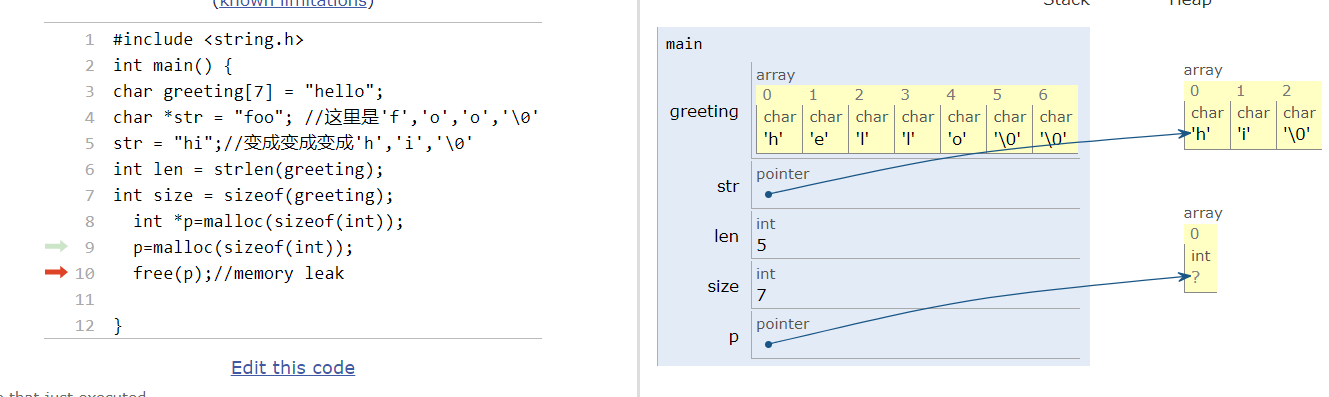
Group of answer choices Data, Stack, Heap, Code Heap, Stack, Code, Data Stack, Data, Code, Heap Stack, Heap, Data, Code Code, Data, Heap, Stack

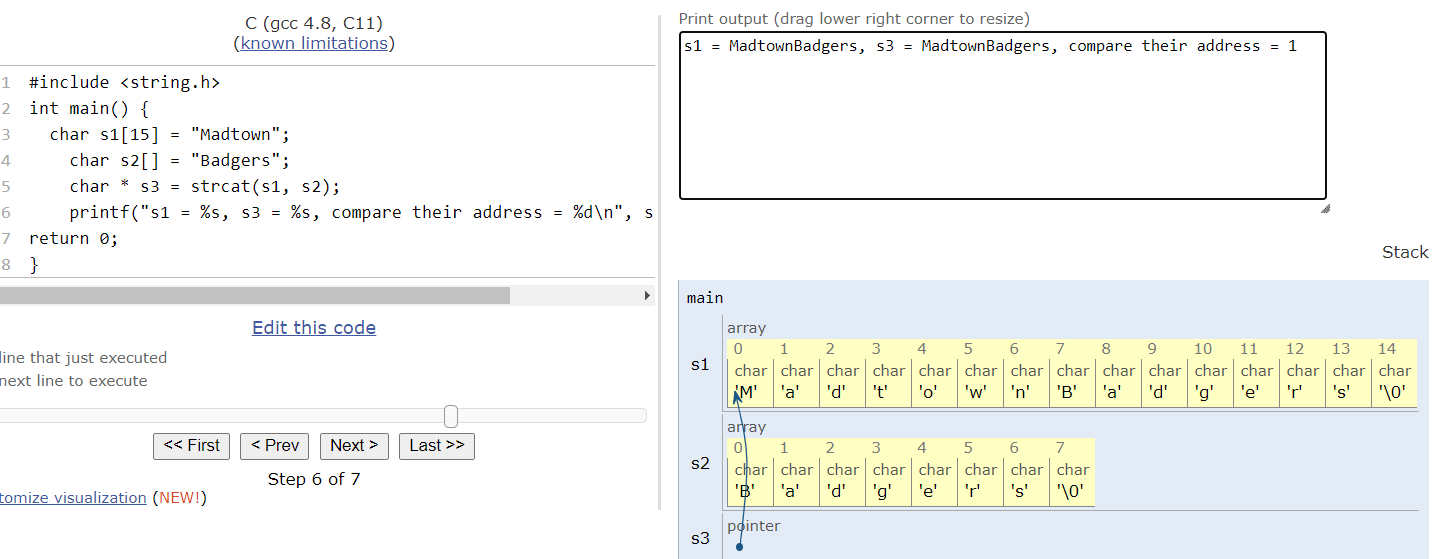


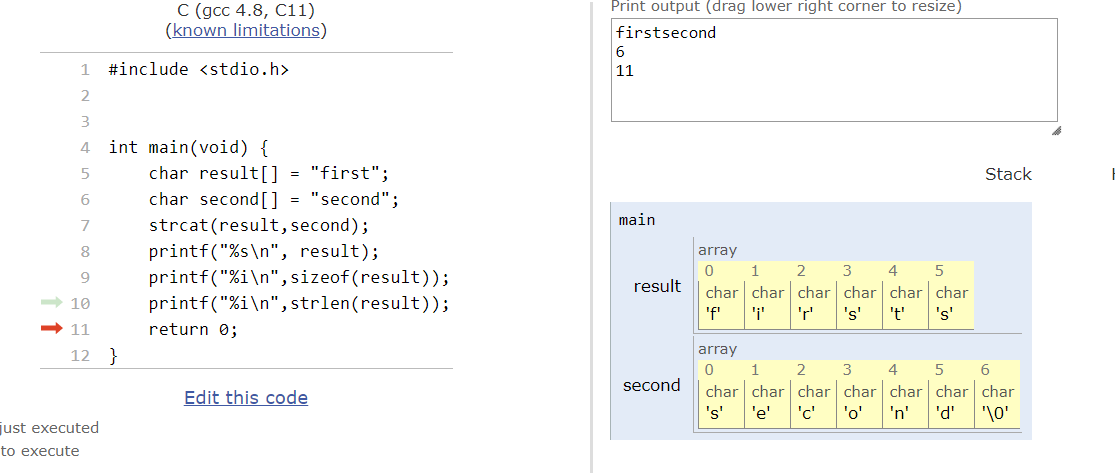




选2 3 4







false fragment:adjacent free block

external fragment: non-adjacent free + adjacent free