NTNU

Candidate Number:

EXAM

COURSE NAME: Mathematics for Games Programming

COURSE NUMBER: REA2061

EXAM DATE: 13th of June 2016

CLASS: Bachelor Games Programming

TIME: 0900-1400

COURSE CONTACTS: Bernt Tore Jensen (46250024)

NUMBER OF PAGES: 4 (inclusive front page)

AIDS ALLOWED: A: All written and printed material. All calculators.

USE A PEN FOR YOUR ANSWERS.

Remember to put your candidate number on all written pages.

When finished, separate the white and yellow page copies in two folders.

The candidate can keep the blue copy, exam questions and drafts.

There are 5 problems, each contributing 20 percent towards your grade. Necessary calculations and explanations must be included.

Problem 1.

You are throwing five fair dice with faces numbered 1 to 6. Calculate the probability of

a) throwing five sixes.
b) throwing two sixes.
c) not throwing five different numbers.
d) throwing at least two sixes.
e) throwing two sixes and two fives.

Problem 2.

We have two functions

```
Vector2f quadratic(Vector2f P0, Vector2f P1, Vector2f P2, float t);
Vector2f cubic(Vector2f P0, Vector2f P1, Vector2f P2, Vector2f P3, float t);
```

which computes points (in the plane) on the quadratic and cubic Bezier curves with control points Pi and parameter $t = 0...1$.

a) Implement the two functions above.

b) Assume we have

```
Vector2f P0,P1,P2,P3;
P0 = Vector2f(0,0);
P1 = Vector2f(10,0);
P2 = Vector2f(10,10);
P3 = Vector2f(0,10);
```

Compute the points in the plane that are returned with the two function calls

```
quadratic(P0,P1,P2,0.3);
cubic(P0,P1,P2,P3,0.6);
```

Problem 3.

In this problem you will do bitwise operations on a 16-bit unsigned integer x. The bits of x are numbered 0...15 with 0 the least significant bit.

a) Write code which

i) sets bit number 5 in x and leaves all other bits intact.
ii) clears bit number 7 in x and leaves all other bits intact.
iii) clears all bits except bit number 3 and 11.

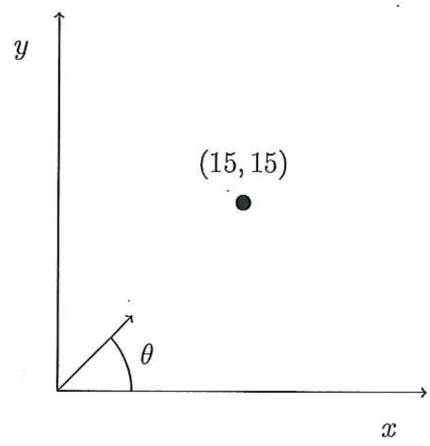
b) Assume we compute

```
y = x | (x+1);
```

i) What is y when $x = 79$?
ii) For an abritrary x: on which bits are x and y equal and on which bits are they different? For which x is x and y equal?

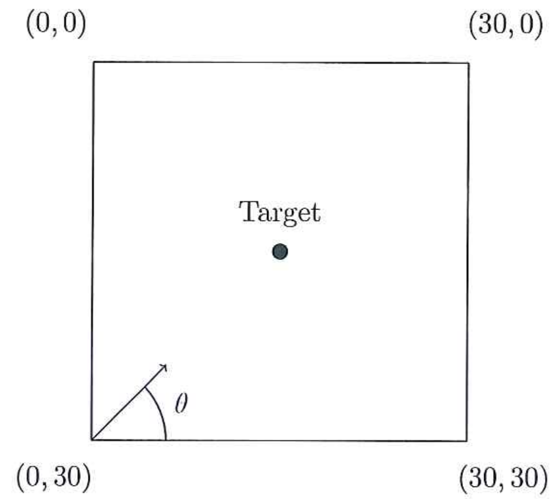
Problem 4.

a) A ball is thrown from the origin with speed 80.



It is supposed to hit the point $(15, 15)$. Find possible angles such that the ball hits the point, or explain why the ball cannot reach the point. You can assume that the acceleration due to gravity is $(0, -10)$ and you can ignore drag.

b) Given a window on the screen with coordinates



Implement the function

```
float findAngle(Vector2f Target);
```

which returns an angle θ which makes the ball thrown from $(0, 30)$ hit the position Target. The speed is 20, acceleration due to gravity is $(0, 10)$ and you can ignore drag. Moreover, the function should return 0 if the target can not be hit.

Problem 5.

Four people come to a river in the middle of night. There is a narrow bridge which can only be crossed by at most two people at a time. It is dark, so they have to use a torch when walking across the bridge. Unfortunately they only have one torch, so the torch needs to be brought back and forth. When two people cross together, they cannot move faster than the slowest of the two.

Person A can cross the bridge in one minute, B in two minutes, C in five minutes, and D in eight minutes.

Find the least amount of time needed for the four people to cross the bridge. For full score you need to have an explicit solution together with a convincing explanation as to why there are no solutions which require less time.

There are 5 problems, each contributing 20 percent towards your grade. Necessary calculations and explanations must be included.

Problem 1.

You are throwing five fair dice with faces numbered 1 to 6. Calculate the probability of

a) throwing five sixes.
b) throwing two sixes.
c) not throwing five different numbers.
d) throwing at least two sixes.
e) throwing two sixes and two fives.

Answers:

I grade all five questions equally.

a) $\frac{1}{6^5} \approx 0.01286\%$.
b) $\frac{C(5,2)\cdot 5^3}{6^5} \approx 16.075\%$
c) $1 - \frac{6\cdot 5\cdot 4\cdot 3\cdot 2}{6^5} \approx 90.74\%$
d) $1 - \frac{5^5 + 5\cdot 5^4}{6^5} \approx 19.624\%$
e) $\frac{C(5,2)\cdot C(3,2)\cdot 4}{6^5} \approx 1.543\%$.

Problem 2.

We have two functions

```
Vector2f quadratic(Vector2f P0, Vector2f P1, Vector2f P2, float t);
Vector2f cubic(Vector2f P0, Vector2f P1, Vector2f P2, Vector2f P3, float t);
```

which computes points (in the plane) on the quadratic and cubic Bezier curves with control points Pi and parameter $t = 0...1$.

a) Implement the two functions above.

b) Assume we have

```
Vector2f P0,P1,P2,P3;
P0 = Vector2f(0,0);
P1 = Vector2f(10,0);
P2 = Vector2f(10,10);
P3 = Vector2f(0,10);
```

Compute the points in the plane that are returned with the two function calls

```
quadratic(P0,P1,P2,0.3);
cubic(P0,P1,P2,P3,0.6);
```

Answers:

I grade with a) and b) both counting 50% towards the score for this problem.

a)

Any language (or pseudo-code) is acceptable. Using math functions such as the pow function in C++ is also fine. Below I assume that Vector2f has an overloaded * operation for scalar multiplication. A solution where the cubic bezier calls the quadratic is considered inefficient, and will result in a 90% correct answer.

Possible implementations are:

```
Vector2f quadratic(Vector2f P0, Vector2f P1, Vector2f P2, float t)
{
      return (1-t)*(1-t)*P0+2*(1-t)*t*P1+t*t*P2;
}
```

```
Vector2f cubic(Vector2f P0, Vector2f P1, Vector2f P2, Vector2f P3, float t)
{
      return (1-t)*(1-t)*(1-t)*P0+3*(1-t)*(1-t)*t*P1+3*(1-t)*t*t*P2+t*t*t*P3;
}
```

b)

$0.7 \cdot 0.7 \cdot (0,0) + 2 \cdot 0.7 \cdot 0.3 \cdot (10,0) + 0.3 \cdot 0.3 \cdot (10,10) = (4.2,0) + (0.9,0.9) = (5.1,0.9)$.

and

$0.4 \cdot 0.4 \cdot 0.4 \cdot (0,0) + 3 \cdot 0.4 \cdot 0.4 \cdot 0.6 \cdot (10,0) + 3 \cdot 0.4 \cdot 0.6 \cdot 0.6 \cdot (10,10) + 0.6 \cdot 0.6 \cdot 0.6 \cdot (0,10) = (7.2, 6.48)$.

Problem 3.

In this problem you will do bitwise operations on a 16-bit unsigned integer x. The bits of x are numbered 0...15 with 0 the least significant bit.

a) Write code which

i) sets bit number 5 in x and leaves all other bits intact.
ii) clears bit number 7 in x and leaves all other bits intact.
iii) clears all bits except bit number 3 and 11.

b) Assume we compute

```
y = x | (x+1);
```

i) What is y when $x = 79$?
ii) For an abritrary x: on which bits are x and y equal and on which bits are they different? For which x is x and y equal?

Answers:

I grade with both a) and b) counting 50% towards the score for this problem.

We will use C++ below, but any language is fine. If the syntax differs from C++, explanations should have been included.

a) i)

```
x = x | (1 << 5);
```

ii)

```
x = x & (~(1 << 7));
```

iii)

```
x = x & ((1<<3) | (1<<11));
```

b) i)

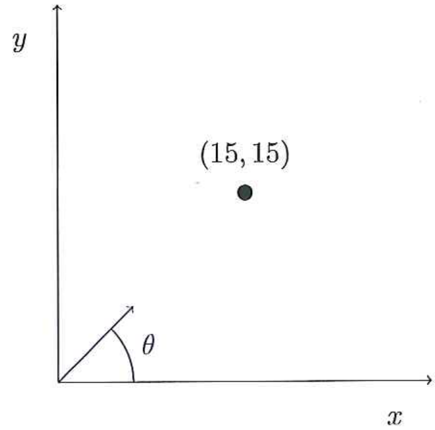
79 is 1001111 in binary and 80 is 1010000 which gives the answer 101111 in binary and $y = 95$ in decimal.

ii)

The two integers x and y are equal on all bits except the rightmost zero-bit in x which is set to 1 in y. Also, x and y are equal on all bits if x has no zero bits, in which case $x = 0xFFFF = 65535$.

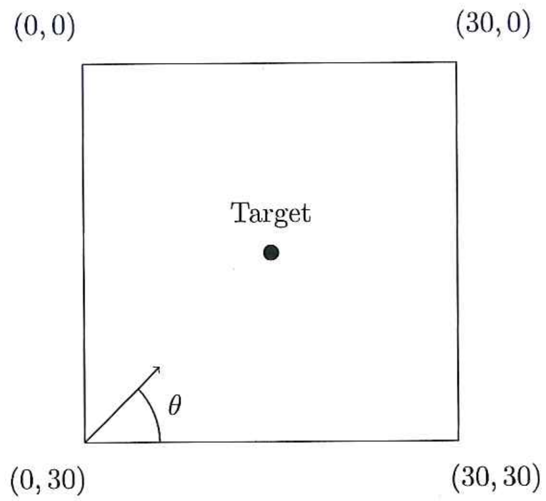
Problem 4.

a) A ball is thrown from the origin with speed 80.



It is supposed to hit the point $(15, 15)$. Find possible angles such that the ball hits the point, or explain why the ball cannot reach the point. You can assume that the acceleration due to gravity is $(0, -10)$ and you can ignore drag.

b) Given a window on the screen with coordinates



Implement the function

`float findAngle(Vector2f Target);`

which returns an angle θ which makes the ball thrown from $(0, 30)$ hit the position Target. The speed is 20, acceleration due to gravity is $(0, 10)$ and you can ignore drag. Moreover, the function should return 0 if the target can not be hit.

Answers:

I grade with a) and b) both counting 50% towards the score for this problem. Any language or pseudo-code is acceptable.

a)

We use the formula

$$\theta = arctan(\frac{v^2 \pm \sqrt{v^4 - g(gx^2 + 2yv^2)}}{gx})$$

where $v = 80$ and $g = 10$ and $(x, y) = (15, 15)$. i.e.

$$\theta = arctan(\frac{80^2 \pm \sqrt{80^4 - 10(10 \cdot 15^2 + 2 \cdot 15 \cdot 80^2)}}{10 \cdot 15})$$

$$= arctan(\frac{80^2 \pm 6246.398963}{150})$$

with solutions $\theta \approx 45.68$ or $\theta \approx 89.32$.

b)

```
float findAngle(Vector2f Target)
{
    // tt is the transformed target
    Vector2f tt = Target;

    // We translate (0,30) to the origin
    tt.y = tt.y - 30;

    // Change sign due to window orientation
    tt.y = -tt.y;

    // calculate the discriminant
    float disc = 20*20*20*20 - 10 * (10 * tt.x * tt.x + 2 * tt.y * 20 * 20);
    if(disc < 0) return 0;    // if no solution
    else return atan((20*20 + sqrt(disc))/(10*tt.x)); // return one solution
}
```

Problem 5.

Four people come to a river in the middle of night. There is a narrow bridge which can only be crossed by at most two people at a time. It is dark, so they have to use a torch when walking across the bridge. Unfortunately they only have one torch, so the torch needs to be brought back and forth. When two people cross together, they cannot move faster than the slowest of the two.

Person A can cross the bridge in one minute, B in two minutes, C in five minutes, and D in eight minutes.

Find the least amount of time needed for the four people to cross the bridge. For full score you need to have an explicit solution together with a convincing explanation as to why there are no solutions which require less time.

Answer:

We do the following:

1. A and B cross: 2 minutes
2. A comes back: 1 minute
3. C and D cross: 8 minutes
4. B comes back: 2 minutes
5. A and B cross: 2 minutes

In total this is 15 minutes.

Another solution is

1. A and B cross: 2 minutes
2. B comes back: 2 minute
3. C and D cross: 8 minutes
4. A comes back: 1 minutes
5. A and B cross: 2 minutes

Also 15 minutes.

Why is this the least amount of time? Here is an explanation:

We will need to do at least three trips across the river and two return trips back again in order for all of them to cross.

If C and D go separate it will require $5 + 8 = 13$ minutes just to get the two of them across. We cannot make three more trips across in just 1 minute. Therefore we need only consider cases where C and D go together.

If we send C and D first, then either C or D will need to come back which leaves us with at most 1 minute to get either ABC or ABD across, which is not possible in three crossings.

If we send C and D last, then either C came back to collect D, or D came back to collect C, which means we were able to move ABD or ABC across in 1 minute. This is again impossible in three crossings.

So C and D has to go on the third crossing. That means AB went first as in the two solutions above. Then after the third trip across the river, either ACD or BCD are on the far side with 12 minutes spent. If either C or D go back we are above 15. The only remaining options are the two solutions above. As the solutions above each use 15 minutes, we can now conclude that there are no solutions which use 14 minutes or less.