

Institutionen för informationsteknologi

TENTAMEN

Kurs: Spelprogrammering 1

Examinationsmoment: Salstentamen

Kurskod: IT148G

Högskolepoäng för examinationsmom.: 3hp

Datum: 2025-01-10

Tentamenstid: 14:15 - 19:30

Ansvarig lärare: Peter Sjöberg

Berörda lärare: Cecilia Wallin

Hjälpmedel/bilagor: Inga

Övrigt:

- Anvisningar:
- ☐ Ta nytt blad för varje lärare
 - ☒ Ta nytt blad för varje ny fråga
 - ☒ Skriv endast på en sida av papperet.
 - ☒ Skriv namn och personnummer på samtliga inlämnade blad.
 - ☒ Numrera lösbladen löpande.
 - ☒ Använd inte röd penna.
 - ☒ Markera med kryss på omslaget vilka uppgifter som är lösta.

Poänggränser:

G 18-32 poäng

U 0-17 poäng

Skrivningsresultat bör offentliggöras inom 18 arbetsdagar

Lycka till!

Fråga 1 (5 poäng)

Koden nedan hanterar ålder på föremål. Klassen `ItemHandler` instantieras vid minst ett tillfälle och funktionen `AgeItems` kallas med jämna mellanrum. När koden körs uppstår ett fel och programmet stannar.

- Beskriv vilket fel som uppstår och vilka förutsättningar som behöver uppfyllas för att felet ska uppstå.
- Förklara hur felet kan åtgärdas.

```
class ItemHandler
{
    private Item[] items;

    public ItemHandler()
    {
        items = new Item[12];
        Random random = new Random();
        for (int i = 0; i <= items.Length; i++)
        {
            items[i] = new Item();
            items[i].age = random.Next() % 58;
        }
    }

    public void AgeItems()
    {
        foreach (Item item in items)
        {
            item.age += 1;
        }
    }
}
```

```
class Item
{
    public int age;
}
```

Fråga 2 (4 poäng)

Skriv C#-kod som skriver ut följande sekvens när det körs:

0,3,1,4,2,5,0,6,1,7,2,8,0,9,1,10,

Det räcker med att skriva koden som behöver vara inuti funktionen `Main` d.v.s. du behöver inte skriva funktioner och klasser. Efter att sekvensens har skrivits ut ska programmet avslutas korrekt. Koden behöver vara rimligt effektiv.

Fråga 3 (3 poäng)

Beskriv hur datastrukturerna `List` och `LinkedList` skiljer sig åt i C# (.NET).

Fråga 4 (4 poäng)

I "pixelbaserade" 2D-spel kan animation hanteras med hjälp av en bild som brukar kallas *spritesheet* eller *tileset*.

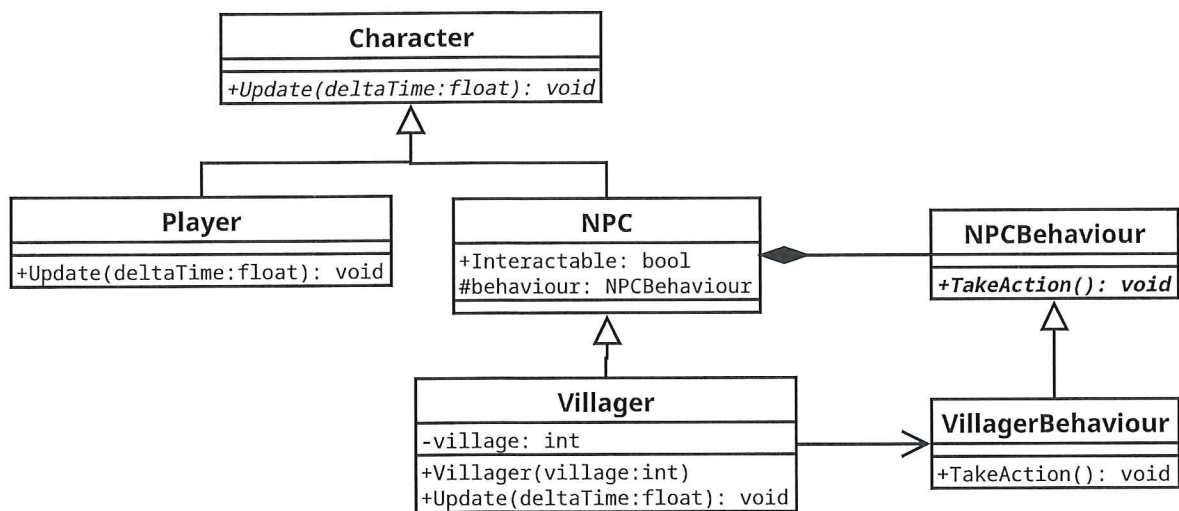
- Beskriv hur bilden är strukturerad och vad den innehåller.
- Beskriv hur bilden används i kod för att skapa en animation, exempelvis en karaktär som går eller en fackla som brinner.

Fråga 5 (5 poäng)

Tolka nedanstående diagram som visar delar av spelfigurshantering. Du kan anta att trianglar kan vara ifyllda.

- Skriv C#-kod för klasserna Character, NPC och NPCBehaviour. Koden ska innehålla alla delar utom kod inuti funktioner.
- Skriv kod för konstruktorn till klassen NPC.

Om du behöver göra antaganden, skriv med dessa i ditt svar som kommentarer.



Fråga 6 (4 poäng)

Nedan finns delar av kod som hanterar ett grafiskt gränssnitt. I en tänkt ändring i koden ska det bli möjligt att samla objekt av typerna Button och Text i en och samma lista. Visa och förklara de ändringar i koden som är nödvändiga för att ovanstående ska bli möjligt.

Du behöver inte visa ändringar i koden så som att hantera listan i klassen GUI, det räcker att behandla förutsättningarna i ditt svar. Du behöver alltså inte skriva fungerande kod i någon del av ditt svar men kan med fördel ange lösa kodrader för att förtydliga din förklaring.

```
class Button
{
    private Vector2i position = new Vector2i(0, 0);
    private string text = "";
    private Color color = Color.Black;
    private Color borderColor = Color.White;
    private bool enabled = true;

    public void Render(Window window)
    {
        // (Kod som ritar knappen)
    }
}

class Text
{
    private Vector2i position = new Vector2i(0, 0);
    private string text = "";
    private Color color = Color.Black;

    public void Render(Window window)
    {
        // (Kod som ritar texten)
    }
}

class GUI
{
    private List<Button> buttons = new List<Button>();
    private List<Text> texts = new List<Text>();

    public GUI()
    {
        buttons.Add(new Button());
        texts.Add(new Text());
    }

    public void RenderAll(Window window)
    {
        foreach (Button b in buttons)
            b.Render(window);

        foreach (Text t in texts)
            t.Render(window);
    }
}
```

Fråga 7 (4 poäng)

I språket C# används nyckelorden `break` och `continue` för att styra flödet i koden. Beskriv skillnaden mellan dessa och ge ett exempel vardera för när de kan användas.

Fråga 8 (3 poäng)

I ett tänkt rollspel med flera spelare vill du hålla objekt som lagrar information om en aktiv besvärjelse. (besvärjelser är aktiva under en viss tid). Informationen lagras bland annat i nedanstående medlemsvariabler:

- Vilken spelare som kastade besvärjelsen
- Typ av besvärjelse
- Hur länge besvärjelsen har varit aktiv

För varje medlem ovan, ge förslag på vilken datatyp som är lämplig att använda och motivera varför datatypen är lämplig. Du behöver göra antaganden för att ge lämpliga förslag, skriv med dessa i ditt svar.