**1. (0.200 Point)**

**Mục tiêu của tái cấu trúc là gì?**

**A. Cải tiến tính dễ hiểu của đoạn mã**

B. Cải tiến cấu trúc đoạn mã

C. Loại bỏ những đoạn mã không cần thiết

D. Tất cả ý trên

**2. (0.200 Point)**

**Tại sao chúng ta cần làm việc nhóm?**

**A. Các dự án phần mềm quá lớn; đòi hỏi tập kỹ năng và vai trò**

B. Các lập trình viên có kinh nghiệm sẽ dạy cho các lập trình viên ít kinh nghiệm

C. Để đánh giá

D. Để ngăn chặn lỗi

**3. (0.200 Point)**

**Phát biểu nào sau đây KHÔNG là tuyên ngôn của Agile?**

A. Phản hồi với thay đổi hơn là bám sát kế hoạch.

**B. Quy trình và công cụ hơn là cá nhân và tương tác.**

C. Cộng tác với khách hàng hơn là đàm phán hợp đồng.

D. Phần mềm chạy tốt hơn là tài liệu đầy đủ.

**4. (0.200 Point)**

**Nếu bạn muốn biết Mô đun 1 hoạt động cùng với Mô đun 2 có đúng hay không. Bạn sẽ thực hiện mức độ kiểm thử nào?**

A. Kiểm thử đơn vị

B. Kiểm thử hệ thống

**C. Kiểm thử tích hợp**

D. Kiểm thử chấp nhận

**5. (0.200 Point)**

**Phát biểu nào SAI khi nói về tái cấu trúc**

A. Tái cấu trúc không sửa lỗi, khuyết điểm

B. Tái cấu trúc không thêm chức năng mới

**C. Tái cấu trúc không làm đoạn mã tốt hơn**

D. Tái cấu trúc là kỹ thuật để cải thiện đoạn mã

**6. (0.200 Point)**

**Các mức độ kiểm thử được thực hiện theo thứ tự nào dưới đây?**

**A. Kiểm thử đơn vị, kiểm thử tích hợp, kiểm thử hệ thống, kiểm thử chấp nhận**

B. Kiểm thử tích hợp, kiểm thử đơn vị, kiểm thử hệ thống, kiểm thử chấp nhận

C. Kiểm thử đơn vị, kiểm thử hệ thống, kiểm thử tích hợp, kiểm thử chấp nhận

D. Kiểm thử đơn vị, kiểm thử tích hợp, kiểm thử chấp nhận

**7. (0.200 Point)**

**Quy trình kiểm tra (inspection process) gồm mấy giai đoạn?**

A. 4

B. 5

**C. 6**

D. 7

**8. (0.200 Point)**

**Trong quy trình kiểm tra (inspection process), vai trò nào chịu trách nhiệm sắp xếp các cuộc họp?**

A. Tác giả (author)

B. Người kiểm tra (inspector)

**C. Người điều phối (moderator)**

D. Người viết (recorder)

**9. (0.200 Point)**

**Critical Path là:**

A. Đường tới hạn

B. Đường găng

**C. Đường có tổng thời gian dài nhất**

D. Tất cả đều đúng

**10. (0.200 Point)**

**Phương pháp phát triển phần mềm linh hoạt Agile có bao nhiêu nguyên tắc?**

A. 10

B. 11

**C. 12**

D. 13

**11. (0.200 Point)**

**Ý nghĩa cơ bản của đường găng:**

A. Cho biết thời gian tối thiểu để hoàn thành dự án

B. Cho biết thời gian tối đa để hoàn thành dự án

**C. Cho biết mối quan hệ phụ thuộc giữa các công việc**

D. Cho biết tổng chi phí mà doanh nghiệp phải trả

**12. (0.200 Point)**

**Kiểm thử hệ thống (System testing) được thực hiện ngay sau:**

A. Kiểm thử đơn vị

B. Kiểm thử chấp nhận

**C. Kiểm thử tích hợp**

D. Kiểm thử module

**13. (0.200 Point)**

**Các hoạt động phát triển phần mềm theo Scrum diễn ra trong một chu trình ngắn và lặp lại, gọi là gì?**

A. Increment

B. Iteration

**C. Sprint**

D. Cycle

**14. (0.200 Point)**

**Mục đích cuộc họp Daily Scrum là gì?**

**A. Để các thành viên trong đội Scrum chia sẻ với nhau tiến độ công việc và khó khăn nếu có**

B. Để Scrum Master giao việc cho các Developers

C. Để Developers báo cáo công việc cho Scrum Master

D. Để Developers báo cáo công việc cho Product Owner

**15. (0.200 Point)**

**Chọn phát biểu SAI về WBS**

A. WBS cung cấp cơ sở để lập kế hoạch và quản lý các lịch biểu, chi phí và những biến đổi dự án

B. WBS là cây phân cấp các nhóm công việc cần thực hiện trong dự án, nó xác định phạm vi tổng thể của dự án

C. WBS là cần thiết khi quản lý dự án IT

**D. WBS là không cần thiết khi quản lý dự án IT**

**1. (0.200 Point)**

**Quy trình phần mềm mà trong đó tất cả các hoạt động được lên kế hoạch trước và tiến độ được đo lường dựa trên kế hoạch này được gọi là quy trình gì?**

**A. Plan-driven process**

B. Agile process

C. Component-based process

D. Reuse-oriented process

**2. (0.200 Point)**

**Mô hình thác nước (Waterfall model) là cách tiếp cận TỐT NHẤT khi triển khai các phần mềm như thế nào?**

A. Khi muốn thêm một số tính năng mới vào hệ thống

B. Khi hệ thống có yêu cầu không ổn định

**C. Khi hệ thống phần mềm có quy mô nhỏ và yêu cầu không rõ ràng**

D. Khi hệ thống phần mềm lớn và được phát triển bởi các đội nhóm làm việc ở nhiều địa điểm khác nhau

**3. (0.200 Point)**

**Từng đơn vị riêng lẻ của chương trình hoặc các lớp đối tượng sẽ được kiểm tra. Loại kiểm thử này tập trung vào kiểm tra chức năng của các đối tượng và phương thức hoạt động của chúng. Loại kiểm thử này được gọi là gì?**

**A. Kiểm thử các đơn vị - Unit testing**

B. Kiểm thử các phân hệ - Component testing

C. Kiểm thử hệ thống - System testing

D. Kiểm thử chấp nhận - Acceptance testing

**4. (0.200 Point)**

**Nguyên tắc TỐT NHẤT của việc lập lịch trình cho dự án phần mềm (Project Scheduling) là gì?**

A. Tất cả các tác vụ được tổ chức thực hiện đồng thời và song song với nhau

B. Tối đa hóa sự phụ thuộc giữa các tác vụ để tránh sự chậm trễ do có một tác vụ đang chờ một tác vụ khác hoàn thành

C. Tất cả các tác vụ đều được sắp xếp độc lập và không phụ thuộc lẫn nhau

**D. Giảm thiểu sự phụ thuộc giữa các tác vụ để tránh sự chậm trễ do có một tác vụ đang chờ một tác vụ khác hoàn thành**

**5. (0.200 Point)**

**Thông thường thì thứ tự của các hoạt động kiểm thử phần mềm được diễn ra như thế nào?(1)Kiểm thử tích hợp (Integration testing) (2) Kiểm thử đơn vị (Unit testing) (3) Kiểm thử hệ thống (System testing) (4) Kiểm thử chấp nhận (Acceptance testing)**

A. 1 => 2 => 3 => 4

B. 4 => 3 => 2 => 1

**C. 2 => 1 => 3 => 4**

D. 4 => 1 => 3 => 2

**6. (0.200 Point)**

**Mô hình quy trình phần mềm nào dựa trên việc tái sử dụng một cách có hệ thống, trong đó hệ thống phần mềm sẽ được tích hợp từ các thành phần có sẵn hoặc từ các thành phần thương mại hóa COTS (Commercial-off-the-shelf)**

A. Waterfall model

B. Incremental model

C. Reuse-oriented model

**D. Component-based model5**

**7. (0.200 Point)**

**Phát biểu nào là SAI khi nói về Kiểm thử phần mềm (Testing)**

A. Kiểm thử là hoạt động nhằm kiểm tra những gì mà một chương trình dự định sẽ thực hiện nhằm phát hiện ra các lỗi của chương trình trước khi nó được đưa vào sử dụng.

**B. Khi kiểm thử phần mềm, ta cho chạy thử chương trình sử dụng bộ dữ liệu nhân tạo.**

C. Khi kiểm thử phần mềm, ta phát hiện sự bất thường trong các thuộc tính chức năng và cả các thuộc tính phi chức năng của hệ thống.

D. Kiểm thử nhằm mục đích tìm ra tất cả các khuyết điểm và lỗi của hệ thống

**8. (0.200 Point)**

**Chuỗi các hoạt động nào sau đây là ĐÚNG khi nói về quy trình phần mềm trong mô hình thác nước (Waterfall model)?(1) Thiết kế hệ thống và thiết kế phần mềm (2) Phân tích và định nghĩa yêu cầu (3) Tích hợp và kiểm thử hệ thống (4) Triển khai và kiểm thử đơn vị (5) Vận hành và bảo trì**

A. 1 => 4 => 3 => 2 => 5

**B. 1 => 2 => 3 => 4 => 5**

C. 2 => 1 => 4 => 3 => 5

D. 2 => 3 => 4 => 1 => 5

**9. (0.200 Point)**

**Phát biểu nào ĐÚNG nhất khi nói về Critical Path?**

A. Người quản lý dự án không cần quan tâm đến Critical Path

B. Người quản lý dự án không cần phân bổ những người có kinh nghiệm để thực hiện các nhiệm vụ trên Critical Path

C. Bất kỳ sự chậm trễ nào diễn ra ở các hoạt động trên Critical Path đều không gây ra sự chậm trễ của toàn bộ dự án

**D. Người quản lý dự án nên phân bổ những người có kinh nghiệm để thực hiện các nhiệm vụ trên Critical Path**

**10. (0.200 Point)**

**Điều nào trong các điều sau đây là SAI khi nói về các nguyên tắc cơ bản trong kỹ thuật phần mềm**

A. Hệ thống nên được phát triển thông qua một quy trình được quản lý chặt chẽ

B. Độ tin cậy và hiệu suất hoạt động rất quan trọng đối với tất cả các loại hệ thống

C. Hiểu và quản lý các yêu cầu và đặc điểm kỹ thuật của phần mềm là rất quan trọng

**D. Không nên tái sử dụng các thành phần phần mềm đã được phát triển trước đây mà nên viết phần mềm mới hoàn toàn**

**11. (0.200 Point)**

**Trong quá trình phát triển phần mềm, nhóm thực hiện dự án gặp phải tình huống các thành phần phần mềm được dự định sẽ tái sử dụng nhưng khi tích hợp với hệ thống mới lại phát sinh ra các lỗi không theo kế hoạch ban đầu. Đây được gọi là rủi ro loại gì?**

**A. Rủi ro về công nghệ**

B. Rủi ro về con người

C. Rủi ro về tổ chức

D. Rủi ro về công cụ

**12. (0.200 Point)**

**Quy trình quản lý rủi ro diễn ra theo trình tự các bước như thế nào? (1) Phân tích rủi ro (2) Lập kế hoạch rủi ro (3) Nhận diện rủi ro (4) Giám sát rủi ro**

**A. (3) – (1) – (2) – (4)**

B. (4) – (2) – (3) – (1)

C. (1) – (2) – (3) – (4)

D. (3) – (2) – (1) – (4)

**13. (0.200 Point)**

**Loại kiểm thử nào sau đây không thuộc loại kiểm thử trong giai đoạn phát triển phần mềm (Software Development)?**

A. Kiểm thử các đơn vị - Unit testing

B. Kiểm thử các phân hệ - Component testing

C. Kiểm thử hệ thống - System testing

**D. Kiểm thử chấp nhận - Acceptance testing**

**14. (0.200 Point)**

**Đánh giá khả năng một rủi ro nào đó có thể xảy ra cùng hậu quả và mức độ nghiêm trọng mà rủi ro này gây ra cho dự án phần mềm được gọi là bước …**

**A. Phân tích rủi ro (Risk analysis)**

B. Lập kế hoạch rủi ro (Risk planning)

C. Nhận diện rủi ro (Risk indentification)

D. Giám sát rủi ro (Risk monitoring)

**15. (0.200 Point)**

**Phát biểu nào là SAI khi nói về trách nhiệm nghề nghiệp của các kĩ sư phần mềm**

**A. Kỹ sư phần mềm chỉ cần tập trung vào việc áp dụng các kỹ năng và kỹ thuật để giải quyết các công việc**

B. Các kỹ sư phần mềm phải cư xử một cách trung thực và có trách nhiệm đạo đức như những người làm nghề chuyên nghiệp.

C. Các kỹ sư thông thường nên tôn trọng tính bảo mật của công ty hoặc khách hàng của họ, bất kể thỏa thuận bảo mật chính thức đã được ký kết hay chưa.

D. Kỹ sư phần mềm nên am hiểu luật pháp nước sở tại về việc sử dụng tài sản trí tuệ như bằng sáng chế, bản quyền để đảm bảo rằng tài sản trí tuệ của công ty và của khách hàng

ÔN TẬP

**Trả lời ngắn:**

**1.**

Sơ đồ Gantt Chart và sơ đồ Critical Path biểu diễn những loại thông tin giống và khác nhau như thế nào về lịch trình của một dự án phần mềm?

- Giống nhau: Cả hai sơ đồ đều mô tả thời gian và sự phụ thuộc giữa các tác vụ trong dự án phần

mềm

- Khác nhau:

+ Sơ đồ Gantt Chart hiển thị lịch trình toàn bộ dự án trong một biểu đồ dạng thanh ngang,

cho thấy thời gian bắt đầu và kết thúc của từng tác vụ. Nó cung cấp cái nhìn tổng quan về

lịch trình và thể hiện cùng lúc nhiều tác vụ

+ Sơ đồ Critical Path chỉ tập trung vào các tác vụ quan trọng nhất và thời gian cần thiết

để hoàn thành dự án. Nó chỉ ra tác vụ chiếm vai trò quyết định đối với thời gian dự án và

hiển thị mối quan hệ phụ thuộc giữa các tác vụ trên đường đi quan trọng nhất

**2.**

Liệt kê bốn thuộc tính cơ bản và quan trọng mà tất cả các phần mềm cần có để được đánh giá là tốt?

* Tính chính xác
* Tính hiệu quả
* Tính bảo mật, an toàn
* Sự hài lòng của người sử dụng

**3.**

Quy trình phần mềm (Software Process) bao gồm các hoạt động cơ bản nào?

- Thu thập yêu cầu

- Phân tích yêu cầu

- Thiết kế

- Lập trình

- Kiểm thử

- Triển khai và duy trì

**4.**

Khi lập lịch cho dự án phần mềm, Work Breakdown Structure được dùng để làm gì?

- Phân chia dự án thành các thành phần nhỏ hơn và quản lý chúng dễ dàng hơn

- Xác định các gói công việc và tác vụ cụ thể để thực hiện dự án

- Xác định mức độ phụ thuộc giữa các tác vụ và quyết định thứ tự thực hiện chúng

- Đánh giá và ước lượng nguồn lực, thời gian và ngân sách cần thiết cho từng phần của dự án

**5.**

Quy trình nhanh (Agile process) là quy trình phát triển phần mềm như thế nào?

- Quy trình linh hoạt (Agile process) là một quy trình phát triển phần mềm linh hoạt và có tính tương tác

**6.**

Mô hình thác nước (Waterfall model) là mô hình phát triển phần mềm theo kế hoạch (Plan-driven process) hay là quy trình nhanh (Agile process)? Vì sao?

Mô hình thác nước (Waterfall model) là mô hình phát triển phần mềm theo kế hoạch

(Plan-driven process)

Vì :

- Nó được xây dựng dựa trên việc lập kế hoạch chi tiết và tuân theo một quy trình tuyến tính và tuần tự

- Các giai đoạn và hoạt động được thực hiện theo thứ tự cố định và không có sự chồng chéo giữa

chúng

- Yêu cầu được xác định trước và không có sự thay đổi đáng kể trong quá trình phát triển

- Quy trình không có linh hoạt và khó thích nghi với các yêu cầu thay đổi hoặc phản hồi từ khách

hàng

**7.**

Nêu các đặc điểm của nhóm có hiệu suất cao (High Performance Team).

* Họ có mục tiêu rõ ràng và gắn chặt với các ưu tiên của đội nhóm và tổ chức.
* Họ hiểu công việc của mình có tác động như thế nào đến sứ mệnh của tổ chức.
* Họ xác định rõ mọi vai trò và trách nhiệm.
* Họ có sự rõ ràng và tôn trọng trong giao tiếp.

**8.**

Hãy liệt kê các loại kiểm thử tích hợp (Integration testing).

* Kiểm thử tích hợp từ dưới lên (Bottom-Up Integration Testing)
* Kiểm thử tích hợp từ trên xuống (Top-Down Integration Testing)
* Kiểm thử tích hợp theo big bang (Big Bang Integration Testing)
* Kiểm thử tích hợp hỗn hợp (Sandwich/Hybrid Integration Testing)
* Kiểm thử tích hợp tiến bộ (Incremental Integration Testing)
* Kiểm thử tích hợp chức năng (Functional Integration Testing)

**9.**

Kiểm thử Beta là gì?

Là thủ tục của một sản phẩm phần mềm để thử nghiệm bởi các khách hàng thực trong môi trường thực tế / ảo trước khi phát hành

**10.**

Tại sao cần kiểm thử trong giai đoạn sử dụng của người dùng (User testing)?

Vì :

Kiểm thử trong giai đoạn sử dụng của người dùng giúp đảm bảo sản phẩm phù hợp với nhu cầu thực tế và mong đợi của người dùng. Nó giúp phát hiện các lỗi, cải thiện trải nghiệm, đánh giá tính khả dụng và hiệu suất trong môi trường thực tế. Việc này giảm rủi ro phát hành, tăng độ tin cậy, và xác minh các tính năng hoạt động đúng trên nhiều nền tảng khác nhau.

**Tự luận**

1. **Dùng junit test case hoặc Nunit Test để test các phương thức của lớp HOCVIEN ( 3 điểm)**

public class HOCVIEN {

private String MaHV;

    private  String Hoten;

    private String Ngaysinh;

    private double D1,D2,DT;

    public void setMaHV(String MaHV)

    {

    this.MaHV=MaHV;    }

    public String getMaHV()

    {

    return this.MaHV;    }

    public void setHoten(String Hoten)

    {

    this.Hoten=Hoten;    }

    public String getHoten()

    {

    return this.Hoten;    }

    public void setNgaysinh(String Ngaysinh)

    {

    this.Ngaysinh=Ngaysinh;    }

    public String getNgaysinh()

    {

    return this.Ngaysinh;    }

    public HOCVIEN(String HT,String NS,String Mhv,double d1,double d2,double d3)

    {

this. MaHV =Mhv;

this.Hoten=HT;

       this.Ngaysinh=NS;

       this.D1=d1;

        this.D2=d2;

        this.DT=dt;

    }

    public  double  DTB()

    {

      return ((this.D1+this.D2)/2 + this.DT\*2)/3;     }

    public   String Xeploai()

    {

      double dtb=this.DTB();

      if(dtb<3)

          return "Kem";

      else if(dtb<5)

          return "Yeu";

      else if(dtb<=6)

          return "Trungbinh";

      else if(dtb<8)

          return "Kha";

      else

          return  "Gioi";     }  }

**2. Dùng junit test case hoặc Nunit Test để test các phương thức của lớp CHUNHAT( 3 điểm)**

public class CHUNHAT

{

public int D,R;

public CHUNHAT ()

{

d=1;

r=1;

}

public CHUNHAT (int d, int r)

{

this.D=d;

this.R=r;

}

public int  DienTich()

{

return(this.D\* this.r);

}

public int ChuVi()

{

return((this.D+this.R)\*2);

}

}

**3.    Dùng junit test case hoặc Nunit Test để test các phương thức của lớp HINHTRON( 3 điểm)**

public class HINHTRON {

    public double R;

    public HINHTRON()

    { this.R=0;   }

 public HINHTRON(double r)

    {

        this.R=r;

    }

     public double **Chuvi**()

    {

        return 2\*this.R\*Math.PI;

    }

    public double **Dientich**()

    {

     return this.R\*this.R\*Math.PI;

    }

}

1. **Dùng junit test case hoặc Nunit Test để test các phương thức sau ( 3 điểm)**

             Public double Sum(int A, int B)

              {

                      Return (A+B);

              }

             Public double Average(double A, int B)

              {

                      Return (A+B)/2;

              }

             Public double Multi(double A, int B)

              {

                      Return (A\*B);

              }

             Public double Division(int  A, double B)

              {

                      Return (A/B);

              }