# Test di valutazione – Modulo 1

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | Nome | Maria Chiara |
|  |  | Cognome | Colla |
|  |  | Data | 05/02/2021 |

Leggete attentamente ogni domanda e argomentare quanto più possibile fornendo anche degli esempi.  
ATTENZIONE: Le domande a risposta multipla possono contenere più risposte corrette.

1. *Spiegare le differenze tra memoria principale, secondaria e virtuale.*

*La memoria principale è una memoria:*

* a breve/medio termine: contiene i dati per il tempo necessario all’esecuzione,
* volatile: se la macchina viene spenta durante l’esecuzione i dati presenti nella mememoria vengono persi
* un esempio di memoria principale è la RAM (Random Access Random): che utilizza un acesso random per recuperare i dati, ovvero che il tempo di accesso per ogni cella di memoria è ugale(non dipende dalla sua posizione), in questo modo l’accesso al dato è più rapido.
* Un altro esempio è la cache: memoria principale semi-volatile: contiene i dati che il processore utilizza più spesso e che difficilmente cambiano.

La memoria secondaria e una memoria:

* a lungo termine: contiene i dati per lunghi periodi,
* non è volatile: se la macchina viene spenta durante l’utilizzo non si avrà una perdita di dati
* esegue l’accesso ai dati in modo sequenzione: il tempo impiegato per recuperarli dipende dalla loro posizione in memoria
* un esempio di memoria secondaria: disco rigido, hard-disk, cd/dvd

La memoria virtuale: è un meccanismo, utilizzato dal Sistema Operativo, con il quale l’applicazione vede la capacità (Quantità di memoria che può utilizzare) della RAM molto maggiore rispetto a quella effettiva, con il seguente modo:

* nella memoria principale vengono salvate le istruzioni necessarie nell’immediato,
* nella memoria secondaria quelle che non serveno nell’immediato,
* in questo modo il processore esegue le estrizioni presenti nella memoria principale
* e si ha uno scambio delle informazioni tra le due memorie, detto Swap.

1. *Cosa significa che un linguaggio è tipizzato? Spiegare la suddivisione in tipi in C# e le relative caratteristiche.*

*Linnguaggio tipizziati significa che ogni variabile deve avere in tipo, ovvero il range dei possibili valori che può assure, i possibili tipi in C# sono:*

* *Int: numeri interi, con i quali si possono usare le operazioni aritmentiche che seguono le regole matematiche,*
* *Decimal: numeri decimali, suddivisi in: float, double e decimal (scritti in ordine decrescente), anche loro utilizzano le operazini aritmentiche,*
* *String: sequenze finite di caratteri, hanno dei metodi propri:*
  + *Length: resituisce la lunghezza della stringa*
  + *Trim: restituisce la stringa senza gli ‘spazi’*
  + *Substring: restituisce una sottostringa*
* *DataTime: ritorna la data e l’ora richiesti*

1. *Cos’è un sistema operativo? Dare una spiegazione delle funzionalità.*

*Un Sistema operative è l’insieme di software più importante, serve per gestire l’accesso alle risorse del computer, gestisce:*

* *Il processore, attraverso il kernel: con il quale protegge l’accesso diretto alla memoria, garantisce un esecuzione sicura dei programmi (evita la loro sovrascrizione), permette il multi-tastink(esegucuzione di programmi in contemporanea)*
* *L’interfaccia utente: suddivisa in*
  + *Interfaccia grafica: l’interfaccia utilizzata dall’utente per utilizzare le risorse del computer*
  + *Interfaccia bash: utilizzo e gestione dell’interfaccia utente con programmazione a riga di comando*
  + *Interfaccia vocale: assistente vocale per eseguire le operazioni sull0interfaccia grafica( es: Siri e Cortana)*
* *Memoria: utilizzando e gestendo la memoria virtuale, assegna ad ogni processo una parte di memoria e coordina la loro gestione.*
* *File system: la gestione e organizzazione gerarchica delle informazioni, utilizzando:*
  + *File: contenitori delle informazini*
  + *Cartelle: contenitori dei file*

1. *Descrivere il funzionamento degli operatori logici e delle relative tabelle di verità.*

*Gli operatori logici valutano una espressione e restituiscono un valore binario (0: falso, 1: vero), 3 tipi:*

* *And: l’operatore and restituisce vero (1) solo se entrambi sono veri, altrimenti falso(0):*
  + *0 and 0 = 0 - 0 and 1= 0 - 1 and 0= 0 - 1 and 1= 1*
* *Or: restituisce falso (0) solo se entrambe sono false, altrimenti vero (1):*
  + *0 or 0= 0 - 0 or 1= 1 - 1 or 0= 1 - 1 or 1=1*
* *Not: nega il valore:*
  + *not 1= 0 - not 0 = 1*

1. *Spiegare la differenza tra ricorsione e iterazione e le relative peculiarità.*

*Iterazione: ripete un blocco di codice finchè un espresione viene valutata vera*

* *Ogni iterazione viene aperta e chiusa ogni volta che inizia e finisce*
* *Spreca poco memoria*
* *Più lento con poche iterazioni*

*Ricrosione: all’interno di una funzione si ha una chiamata alla funzione stessa:*

* *Prima di chiudere ogi iterazione si deve aspettare il risultato finitale (rimangono tutte aperte fino alla produzione di un risultato)*
* *Si ha uno spreco di memoria*
* *Più veloce con poche ripetizioni*

1. *Dare una definizione di casting. Fornire degli esempi.*

*Casting: conversion di un tipo con un tipo affine, se non va a buon fine si otterà un eccezione, esistono due tipi di cast:*

* *Cast implicito: non causa la perdita del valore,*
* *Cast esplicito: si potrebbe avere una perdito del valore, si utilizzamo gli operatori di casting*

*Tipi di conversion:*

* *Convert: se non è possibile causa un eccezione, es: float x = Convert.ToFloat(y);*
* *Parse: es: int x = Int32.Parse(y);*
* *As: es: int x = y as int*
* *Is: es: y is int*

*Utilizzati con la struttura try catch per catturare le possibili eccezioni:*

* *Try{...} cath(exception e) {...}*

1. *Spiegare cosa è una routine. Specificare le tipologie di routine e fornire degli esempi per ciascuna.*

*Le routine sono delle istruzioni in un unico blocco di codice che formano delle entità richiamabili, ne esistono di due tipi:*

* *Procedure*
* *Funzioni*

1. *Descrivere la funzione del compilatore*

*Il compilatore esegue il controllo della sintassi dell linguaggio di programmazione e lo traduce in linguaggio macchina.*

*Legge l’intero codice, codice sorgente, e produce un codice eseguibile in linguaggio macchina.*

1. *Descrivere le funzioni di un sistema di versionamento e le diverse tipologie*

*I sistemi di versionamento tengono aggiornate le versioni dei file, e registrano nel tempo i loro cambiamenti, ne esistono di 3 tipi:*

* *Locali: i file e le loro versioni vengono salvate in locale, questo sistema funziona bene quando si lavora da soli*
* *Centralizzati: si utilizza un unico server centrale sul quale vengono salvati i file e le loro versioni, gli utenti recuperano i file dal server per lavorarci, gli svantaggio sono che se il server cade non si può accedere al server e gli utenti non possono lavorare, e se non si era eseguito il back-up si avrà una perdita di dati*
* *Distribuiti: ogni utente copia lo storico delle modifiche dei file in locale, in questo modo può lavorare in autonomia in locale e poi una volta finito può pubblicare le modifiche nel server, così anche se il server cade non si ha perdita di memoria e ogni utente può lavorare.*

*Esercizio Pratico*

Creare una Console Application che gestisca i Task dell’utente.

Per Task viene inteso un oggetto che ha una descrizione, una data di scadenza e un livello di importanza (Basso, Medio, Alto).

L’utente può:

* Vedere i Task inseriti
* Aggiungere un nuovo Task
* Eliminare un Task
* Filtrare i Task per importanza

Requisiti Tecnici:

-Salvare i Task in un file

-Utilizzare adeguatamente il concetto di classe

-Dividere le funzionalità in relative funzioni e procedure

-Commentare

-Mettere una nomenclatura conforme

-Le date di scadenza devono essere posteriori o uguali rispetto alla data di inserimento

-Controllare l’input utente

Opzionale: Utilizzare Enum

Mettere il codice dell’esercizio in un Repository di GitHub.