PIANO DI TEST CLASSE: Pc

|  |  |  |  |
| --- | --- | --- | --- |
| **Precondizioni** | **Descrizione** | **Azioni** | **Risultato** **Atteso** |
|  | Test costruttore  Pc | Costruisce un oggetto di classe Pc con il costruttore, con parameteri(1,”asus”,3),  Invoca i metodi  getIdentificativo()  getNome(), getQuantitaDisponibile().  Confronta col metodo equals  AssertEquals ogni parametro. | getIdentificativo = 1,  getNome=asus, getQuantitaDisponibile=3 |
|  | Test costruttore vuoto | Costruisce un oggetto di classe Pc utilizzando il costruttore vuoto. Invoca i metodi  getIdentificativo()  getNome(), getQuantitaDisponibile().  Creare un nuovo oggetto di tipo Pc.  Confronta col metodo equals  AssertEquals ogni parametro. | getIdentificativo = 0,  getNome=null, getQuantitaDisponibile=0 |
|  | Test  Costruttore di copia | Costruisce un oggetto di tipo Pc con il costruttore con parametri(1,”asus”,3).  Costruisce un nuovo oggetto di tipo Pc con il costruttore copia passando come parametro l’oggetto precedentemente istanziato.  Confronta col metodo equals  AssertEquals l’uguaglianz atra gli oggetti creati.. | P=new Pc(p) |
|  | Test  setIdentificativo | Costruisce un oggetto di tipo Pc utilizzando il costruttore senza parametri.  Invocare setCodiceIdentificativo(1)  e dopo invoca  AssertEquals per confrontare l’identificativo con 1. | getìIdentificativo=1 |
|  | Test setNome | Costruisce un oggetto di tipo Pc utilizzando il costruttore senza parametri.  Invocare setNome(asus)  e dopo invoca AssertTrue con il compareTo tra il nome e asus | getNome=asus |
|  | Test setQuantitaDisponibile | Costruisce un oggetto di tipo Pc utilizzando il costruttore senza parametri.  Invocare setQuantitaDisponibile (3) e dopo invoca  AssertEquals per confrontare la quantita con 3. | getQuantitaDisponibile=3 |
|  | Test  getIdentificativo | Costruisce un oggetto di classe Pc con il costruttore, con parameteri(1,”asus”,3), e dopo invoca  AssertEquals per confrontare l’identificativo con 1. | getìIdentificativo=1 |
|  | Test  getNome | Costruisce un oggetto di classe Pc con il costruttore, con parameteri(1,”asus”,3), e dopo invoca assertEquals per confrontare il nome con asus. | getNome=asus |
|  | Test  getQuantitaDisponibile | Costruisce un oggetto di classe Pc con il costruttore, con parameteri(1,”asus”,3), e dopo invoca assertEquals per confrontare la quantita disponibile con 3. | getQuantitaDisponibile=3 |
|  | Test toString | Costruisce un oggetto di classe Pc con il costruttore, con parameteri(1,”asus”,3), una stringa che dovra poi essere uguale al valore di ritorno del toString.  Dopo invoca assertEquals per confrontare la stringa creata con il toString dell’oggetto. | S=p.toString() |

PIANO DI TEST CLASSE: Nodo

|  |  |  |  |
| --- | --- | --- | --- |
| **Precondizioni** | **Descrizione** | **Azioni** | **Risultato** **Atteso** |
|  | Test costruttore  Nodo | Costruire un oggetto di classe PC con il costruttore vuoto, lo passa come parametro al costruttore Nodo.  Invoca getInfo() con il metodo equals. | getInfo=p  getLink=null |
|  | Test setInfo | Costruisce 2 oggetti della classe PC.  Il primo lo passa come parametro al costruttore del nodo, il secondo lo utilizza come parametro del metodo setInfo.  Verifica l’uguaglianza tramite il metodo equals. | getInfo=p2  getLink=null; |
|  | Test setLink | Costruisce 2 oggetti di classe Pc e 2 oggetti di classe Nodo.  Dopo setta con setlink del primo nodo il reference del secondo nodo.  Verifica L’uguaglianza tramite il metodo True. | getInfo=p1  getLink=n2 |
|  | Test getInfo | Costruisce un oggetto della classe Pc e uno della calsee nodo.  Setta con setLink del nodo il reference dell’oggetto.  Verifica l’uguaglianza tramite il metodo equals. | p=n.getLink() |
|  | TestGetLink | Costruisce 2 oggetto della classe Pc e 2 della calsee nodo.  Setta con setLink del nodo 1il reference dell’oggetto 1 e il 2 con il 2.  Verifica l’uguaglianza tramite il metodo equals. | n2=n1.getLink() |