**Documentație**

1. Enunț

Aplicația gestionează cheltuielile de familie efectuate într-o lună. Pentru o cheltuială se vor reține ziua (din timpul lunii), suma, tipul cheltuielii (5 categorii posibile: mâncare, întreținere, îmbrăcăminte, telefon, altele). Aplicația permite efectuarea repetată a următoarelor operații:

* Adăugare cheltuieli
* Ștergeri
* Căutări
* Rapoarte
* Filtrări
* Back-up-uri

1. Listă de funcționalități

**-pentru adăugarea cheltuielilor:**

* Adaugă o nouă cheltuială
* Validează datele de intrare
* Actualizează o cheltuială existentă

**-pentru ștergere cheltuielilor:**

* Șterge cheltuieli pentru o zi dată
* Șterge cheltuieli pentru un interval de timp dat
* Șterge cheltuieli de un anumit tip

**-pentru căutarea cheltuielilor:**

* Tipărește cheltuieli mai mare decât o sumă dată
* Tipărește cheltuieși efectuate înainte de o zi dată și mai mici decât o sumă dată
* Tipărește cheltuieli de un anumit tip

**-pentru raporate de cheltuieli:**

* Tipărește suma totală pentru un anumit tip de cheltuială
* Găsește ziua în care suma cheltuită e maximă
* Tipărește cheltuieli de o anumită sumă
* Tipărește cheltuieli ordonate în funcție de tipul lor

**-pentru filtrarea cheltuielilor:**

* Elimină cheltuieli de un anumit tip
* Elimină cheltuieli mai mici decât o sumă dată

**-pentru revenirea cheltuielilor anterioare:**

* Reface ultima operație de alterare a cheltuielilor

1. Iterații

|  |  |
| --- | --- |
| **Iterații** | **Funcționalități propuse** |
| I1 | * Interfață utilizator * Validează datele de intrare * Adaugă o nouă cheltuială * Tipărește cheltuieli mai mare decât o sumă dată * Tipărește cheltuieși efectuate înainte de o zi dată și mai mici decât o sumă dată * Tipărește cheltuieli de un anumit tip |
| I2 | * Elimină cheltuieli de un anumit tip * Elimină cheltuieli mai mici decât o sumă dată * Tipărește suma totală pentru un anumit tip de cheltuială * Găsește ziua în care suma cheltuită e maximă * Tipărește cheltuieli de o anumită sumă * Tipărește cheltuieli ordonate în funcție de tipul lor |
| I3 | * Reface ultima operație de alterare a cheltuielilor * Actualizează o cheltuială existentă * Șterge cheltuieli pentru o zi dată * Șterge cheltuieli pentru un interval de timp dat * Șterge cheltuieli de un anumit tip |

1. Lista de taskuri (corespunzătoare iterației 2)
   * **elim\_tip(c)**
     + funcția tipărește toate cheltuielile, mai puțin cele de un anumit tip dat
     + returnează cheltuielile cerute, altfel returnează un mesaj de eroare in cazul în care nu există cheltuieli care îndeplinesc condiția data
     + c - listă, cuprinde cheltuielile cu care se face căutarea si afișarea
   * **test\_elim\_tip()**
   * **elim\_sum(c)**
     + funcția tipărește toate cheltuielile, mai puțin cele mai mici decât o suma dată
     + returnează cheltuielile cerute, altfel returnează un mesaj de eroare in cazul în care nu există cheltuieli care îndeplinesc condiția data
     + c - listă, cuprinde cheltuielile cu care se face căutarea si afișarea
   * **test\_elim\_sum()**
   * **reun\_liste(a, b)**
     + funcția returnează intersecția listelor a și b, cu elemente strict crescătoare
     + a, b - liste
     + returnează o listă cu elementele comune a și b, lista vidă altfel
   * **test\_reun\_liste()**
   * **verif\_cond\_lista(l, d, x, cond, comp)**
     + funcția verifică dacă un element al listei de cheltuieli îndeplinește condițiile specificate prin variabilele cond și comp
     + l - lista, conține cheltuielile
     + x - număr natural, indexul listelor-element, specifică ziua, tipul sau suma unei cheltuieli
     + cond - 0, 1 sau 2, în funcție de tipul de comparație logică dorită
     + comp - număr întreg, în funcție de care se face comparația logică
     + returnează prin d lista indicilor aparținând l care satisfac condițiile
   * **test\_verif\_cond\_lista()**
   * **sum\_tot(c, l)**
     + funcția returnează suma sumelor acelor cheltuieli din c ai căror indecși se regăsesc in l, altfel 0 dacă l e vidă
     + c - lista cheltuielilor
     + l - lista indecșilor
   * **test\_sum\_tot()**
   * **index\_sum\_max(c)**
     + funcția returnează indexul elementului din lista c cu proprietatea că reține suma maximă a vreunei cheltuieli
     + c - lista cheltuielilor
   * **test\_index\_sum\_max()**
   * **rap\_sum\_tip(c)**
     + funcția calculează suma totala pentru au anumit tip de cheltuiala
     + returnează suma cerută, altfel returnează un mesaj de eroare în cazul în care nu există cheltuieli care îndeplinesc condiția dată
     + c - lista cheltuielilor
   * **rap\_zi\_sum\_max(c)**
     + funcția afișează ziua în care s-a efectuat cheltuiala maximă
     + returnează ziua cerută, altfel returnează un mesaj de eroare în cazul în care nu există cheltuieli care îndeplinesc condiția dată
     + c - lista cheltuielilor
   * **rap\_sum(c)**
     + funcția tipărește toate cheltuielile de o anumită sumă dată
     + returnează cheltuielile cerute, altfel returnează un mesaj de eroare în cazul în care nu există cheltuieli care îndeplinesc condiția dată
     + c - lista cheltuielilor
   * **rap\_tip(c)**
     + funcția tipărește toate cheltuielile în funcție de tip, în ordine
     + returnează cheltuielile cerute, altfel returnează un mesaj de eroare în cazul în care nu există cheltuieli care îndeplinesc condiția dată
     + c - lista cheltuielilor
2. Scenarii de testare

* *pentru* eliminarea cheltuielilor de un anumit tip:

|  |  |  |  |
| --- | --- | --- | --- |
| **Pas** | **Utilizator** | **Program** | **Descriere** |
| a |  | “Alegeți tipul cheltuielilor ce NU doriți să fie afișate:  Dați tipul:  1. Mâncare - tasta 1  2. Întreținere - tasta 2  3. Îmbrăcăminte- tasta 3  4. Telefon - tasta 4  5. Altele - tasta 5  Tipul:” | Se așteaptă introducerea tipului dorit |
| b | 2 |  | Se introduce tipul dorit |
| c |  | “Toate cheltuielile, în afara celor de tipul 2 , sunt:  Ziua 2  Tipul 3  Suma 20.0    Ziua 4  Tipul 5  Suma 30.0    Ziua 1  Tipul 1  Suma 45.0    Ziua 20  Tipul 3  Suma 26.0” | Se afișează toate cheltuielile, în afară de cele cu tipul specificat |
| b | 4 |  | Se introduce tipul dorit |
| c |  | “Toate cheltuielile, în afara celor de tipul 4 , sunt:  Nu au fost găsite astfel de cheltuieli” | Se constată lipsa cheltuielilor dorite (toate cheltuielile sunt de tipul specificat de utilizator) |

* *pentru* eliminarea cheltuielilor mai mici decât o sumă dată:

|  |  |  |  |
| --- | --- | --- | --- |
| **Pas** | **Utilizator** | **Program** | **Descriere** |
| a |  | ”Alegeți suma minimă a cheltuielilor ce NU doriți să fie afișate:  Dați suma (mai mare decât 0):” | Se așteaptă introducerea sumei dorite |
| b | 30 |  | Se introduce suma |
| c |  | “Toate cheltuielile, în afara celor mai mici de suma 30.0 , sunt:  Ziua 4  Tipul 5  Suma 30.0    Ziua 1  Tipul 1  Suma 45.0    Ziua 2  Tipul 3  Suma 60.0    Ziua 16  Tipul 4  Suma 50.0” | Se afișează cheltuielile cu suma mai mare decât sau egală cu suma introdusă de utilizator |
| b | 70 |  | Se introduce suma |
| c |  | “Toate cheltuielile, în afara celor mai mici de suma 70.0 , sunt:  Nu au fost găsite astfel de cheltuieli” | Se constată lipsa cheltuielilor care satisfac condiția (suma >= 70) |

* *pentru* tipărirea sumei totale pentru un anumit tip de cheltuială:

|  |  |  |  |
| --- | --- | --- | --- |
| **Pas** | **Utilizator** | **Program** | **Descriere** |
| a |  | ” Alegeți tipul de cheltuială:    Dați tipul:  1. Mâncare - tasta 1  2. Întreținere - tasta 2  3. Îmbrăcăminte- tasta 3  4. Telefon - tasta 4  5. Altele - tasta 5  Tipul:” | Se așteaptă introducerea tipului dorit |
| b | 5 |  | Se introduce suma |
| c |  | “Suma totală este:  Suma 30.0” | Se afișează suma totală pentru un anumit tip de cheltuială |
| b | 2 |  | Se introduce suma |
| c |  | “Suma totală este:  Nu au fost găsite astfel de cheltuieli” | Se constată lipsa cheltuielilor dorite (nu există cheltuieli de tipul celui specificat de utilizator) |

* *pentru* găsirea zilei în care s-a efectuat cheltuiala de sumă maximă:

|  |  |  |  |
| --- | --- | --- | --- |
| **Pas** | **Utilizator** | **Program** | **Descriere** |
| a |  | ” Ziua în care s-a efectuat cheltuiala maximă este: ” | Se parcurge lista cheltuielilor și se caută suma maximă, respectiv ziua corespunzătoare acesteia |
| b |  | ” Ziua 2  Tipul 3  Suma 60.0” | Se afișează ziua dorită |
| b | 2 |  | Se introduce suma |
| c |  | “Suma totală este:  Nu au fost găsite astfel de cheltuieli” | Se constată lipsa cheltuielilor dorite (nu există cheltuieli de tipul celui specificat de utilizator) |

* *pentru* tipărirea cheltuielilor de o anumită sumă:

|  |  |  |  |
| --- | --- | --- | --- |
| **Pas** | **Utilizator** | **Program** | **Descriere** |
| a |  | ” Alegeți suma cheltuielilor:    Dați suma (mai mare decât 0): ” | Se așteaptă introducerea sumei dorite |
| b | 30 |  | Se introduce suma |
| c |  | “Cheltuielile de suma 30.0 sunt:  Ziua 4  Tipul 5  Suma 30.0” | Se afișează toate cheltuielile a căror sumă este egală cu cea introdusă de utilizator |
| b | 5 |  | Se introduce suma |
| c |  | “Cheltuielile de suma 5.0 sunt:    Nu au fost găsite astfel de cheltuieli” | Se constată lipsa cheltuielilor dorite (nu există cheltuieli a căror sumă să fie 5) |

* *pentru* tipărirea cheltuielilor în funcție de tip, în ordine:

|  |  |  |  |
| --- | --- | --- | --- |
| **Pas** | **Utilizator** | **Program** | **Descriere** |
| a |  | “Tipul 1 :    Ziua 1  Tipul 1  Suma 45.0    Tipul 3 :    Ziua 2  Tipul 3  Suma 20.0    Ziua 2  Tipul 3  Suma 60.0    Ziua 20  Tipul 3  Suma 26.0    Tipul 4 :    Ziua 16  Tipul 4  Suma 50.0    Tipul 5 :    Ziua 4  Tipul 5  Suma 30.0” | Se afișează toate cheltuielile, ordonate după tip |