



TEST PRACTIC 1 – ELEMENTE DE INFORMATICĂ MOBILĂ

Numele și Prenumele Studentului _____
Cont GitHub _____ Grupa _____

Indicații

- Timpul de lucru efectiv este de 60 de minute.
- Punctajul care poate fi obținut pentru acest test practic este de 100 de puncte.
- Acest test practic conține 4 grupuri de subiecte. Este recomandat să fie rezolvate în ordinea în care apar.
- Pentru fiecare subiect poate fi acordat și punctaj parțial, însă exclusiv în condițiile unei implementări funcționale.
- După fiecare 20 de minute, cadrul didactic va evalua măsura în care au fost rezolvate grupurile de exerciții, punctându-le corespunzător.
- La expirarea timpului de lucru, codul sursă trebuie încărcat pe contul Github. Nu se iau în considerare alte modificări locale. Asigurați-vă că după rezolvarea fiecărui exercițiu, acesta este încărcat pe depozitul la distanță!
- Pentru acest test practic puteți consulta orice fel de resursă, dacă aceasta nu implică colaborarea cu alte persoane. Folosirea oricărui mijloc de comunicare se consideră copie și se sancționează cu anularea punctajului aferent testului practic!

SUBIECTE

A.1. [5%] a) În contul Github personal, să se creeze un depozit denumit *PracticalTest01*.

Inițial, acesta trebuie să conțină:

- un fișier `README.md`, în care veți preciza următoarele informații: numele și prenumele, grupa din care faceți parte;
- un fișier `.gitignore` prin intermediul căruia resursele generate ale aplicației Android (clase, fișiere `.ap_`, `.apk`, `.dex`) nu vor fi consemnate în cadrul sistemului de versionare a codului sursă;
- un fișier `LICENSE` care să descrie condițiile de utilizare a aplicației Android (la alegere).

b) Să se descarce pe discul local conținutul depozitului (la distanță) *PracticalTest01*.

c) În mediul integrat de dezvoltare Android Studio, să se creeze un proiect corespunzător unei aplicații Android, având următoarea specificație:

- denumirea aplicației Android și a proiectului este *PracticalTest01*;
- denumirea pachetului corespunzător aplicației Android este `ro.pub.cs.systems.eim.practicaltest01`;
- nivelul minim de API este cel corespunzător versiunii Jelly Bean (4.1) = 16;
- denumirea activității principale a aplicației Android este `PracticalTest01MainActivity`;
- denumirea fișierului ce descrie interfața grafică a activității principale a aplicației Android este `activity_practical_test01_main.xml`.



A.2. [20%] Să se creeze o interfață grafică care conține două câmpuri text, needitabile, dispuse pe o linie, unul lângă celălalt ocupând jumătate din suprafața containerului din care fac parte, având dedesubt câte un buton centrat relativ la ele. În cadrul câmpurilor text, se va afișa o cifră reprezentând numărul de apăsări al butonului (inițial, valoarea va fi 0).



B.1. [10%] Să se asocieze butoanelor un ascultător, astfel încât pentru fiecare eveniment de tip apăsare a butonului, valoarea din câmpul text corespunzător să fie incrementată.

B.2. [15%] **a)** Forțați sistemul de operare Android să nu salveze / restaureze în mod automat starea câmpurilor text în momentul în care aceasta este distrusă din cauza asigurării necesarului de resurse.

b) Implementați un mecanism pentru salvarea / restaurarea valorii din câmpurile text, dacă sistemul de operare Android distruge activitatea spre a asigura necesarul de resurse.

c) Simulați comportamentul de mai sus.

C.1. [15%] Să se implementeze o nouă activitate (`PracticalTest01SecondaryActivity`, având interfața grafică în fișierul `activity_practical_test01_secondary.xml`) în contextul aceleiași aplicații Android, care să poată fi invocată doar prin intermediul unei intenții. Aceasta va conține un câmp text, care va afișa de câte ori au fost accesate cele două butoane (din activitatea principală) în total. Totodată, aceasta va avea și două butoane, *Ok* și *Cancel*, prin care se va reveni la activitatea din care a fost invocată, însă cu rezultate diferite.

C.2. [5%] În activitatea principală, să se creeze un buton prin care se poate invoca cealaltă activitate din aplicația Android, afișând la revenirea din ea (într-un control grafic de tip `Toast`) rezultatul care a fost primit.

D.1. [15%] **a)** Să se implementeze un serviciu de tip `started` denumit `PracticalTest01Service` care primește prin intenția cu care e invocat numerele din câmpurile text și propagă la nivelul sistemului de operare Android un mesaj cu difuzare, la fiecare 10 secunde ce conține data și ora la care a fost transmis, media aritmetică și media geometrică numerelor. Definiți trei acțiuni diferite, care vor fi asociate aleator intenției folosite pentru propagarea mesajului cu difuzare.

b) Să se pornească serviciul atunci când suma valorilor din cele două câmpuri text depășește un prag, definit la nivelul codului sursă sub formă de constantă. Serviciul este oprit (nu mai transmite mesaje cu difuzare) atunci când aplicația Android este distrusă.

D.2. [10%] Să se implementeze un ascultător pentru mesajele de difuzare. Procesarea unui mesaj cu difuzare implică jurnalizarea sa în consolă, folosind o anumită etichetă.

D.3. [5%] Să se încarce rezolvările în depozitul `PracticalTest01` de pe contul Github personal, folosind un mesaj sugestiv.