Egzamin licencjacki/inżynierski — 9 września 2016

Informacja dla zdających egzamin na kierunku informatyka:
Z sześciu poniższych zestawów zadań (Matematyka I, Programowanie, Matematyka dyskretna, Algorytmy i struktury danych, Metody numeryczne, Matematyka II) należy wybrać i przedstawić na osobnych kartkach rozwiązania trzech zestawów.

Informacja dla zdających egzamin na kierunku indywidualne studia informatyczno-matematyczne:
Z sześciu poniższych zestawów zadań (Matematyka I, Programowanie, Matematyka dyskretna, Algorytmy i struktury danych, Metody numeryczne, Języki formalne i złożoność obliczeniowa) należy wybrać i przedstawić na osobnych kartkach rozwiązania trzech zestawów.

Informacja dla wszystkich zdających:
Za brakujące (do trzech) zestawy zostanie wystawiona ocena niedostateczna z urzędu. Egzamin uważa się za zaliczony, jeśli student rozwiąże z oceną dostateczną co najmniej 2 zestawy. Wtedy ocena z egzaminu jest średnią arytmetyczną ocen z trzech wybranych zestawów. Na rozwiązanie przeznacza się czas 3x40=120 minut. Po wyjściu z sali egzaminacyjnej w czasie egzaminu nie ma możliwości powrotu do tej sali i kontynuowania pisania egzaminu.

Matematyka I — Logika dla informatyków

Ile jest funkcji $f : \mathbb{N} \to \mathbb{N}$ spełniających następujące warunki?

$$
\begin{aligned}
f(16) &= 2016 && (1)\\
f(i+j) &= f(i) + f(j) + 1 \quad \text{dla wszystkich } i, j \in \mathbb{N} && (2)
\end{aligned}
$$

Udowodnij poprawność swojej odpowiedzi.

Uwaga: To jest zadanie z logiki. Przy ocenianiu zwrócimy szczególną uwagę na poprawność i klarowność rozumowania.

Programowanie

Za tę część egzaminu można otrzymać 20 punktów. Aby otrzymac ocenę dostateczną, należy zdobyć 7 punktów, próg dla dst+ to 9p, dla db – 11p, dla db+ 13p, dla bdb – 15p.

Zadanie 1. Gramatyka G_1 z symbolem startowym S nad alfabetem $\{a, b\}$ dana jest za pomocą następującego zbioru produkcji:

$$S \to abSb,\ S \to bSab,\ S \to SS,\ S \to \varepsilon$$

Dla gramatyki G przez $L(G)$ rozumieć będziemy język generowany przez G. Dla wyrażenia regularnego r przez $\mathcal{L}(r)$ rozumiemy język opisany przez wyrażenie r.

a) Czy $abbabb$ należy do $L(G_1)$? Odpowiedź uzasadnij. **(1)**

b) Czy gramatyka G_1 jest jednoznaczna? Odpowiedź krótko uzasadnij. **(2)**

c) Podaj gramatykę lub wyrażenie regularne generujące zbiór $L(G_1) \cap \mathcal{L}((ab)^*(aa)^*b^*)$. Odpowiedź krótko uzasadnij. **(3)**

d) Napisz w języku imperatywnym funkcję, która bierze jako wejście napis i zwraca wartość logiczną, równą True wtedy i tylko wtedy, gdy ten napis należy do zbioru $L(G_1) \cap \mathcal{L}(b^*(ab)^*b^*)$ Możesz używać języka wybranego z następującej listy: C, C++, Java, C#, Python, Ruby, PHP, AWK, Pascal. **(4)**

Zadanie 2. (5p) Napisz w Prologu predykat `my_is`, który działa tak, jakby był zdefiniowany w następujący sposób:

```
my_is(V, E) :- V is E.
```

Predykat ten powinien obsługiwać 4 podstawowe działania. W definicji nie możesz korzystać ze standardowego predykatu `is` (i innych arytmetycznych predykatów prologowych), ale możesz używać następujących, zdefiniowanych niżej predykatów:

```
add(X,Y,Z)   :- number(X), number(Y), Z is X+Y.
mult(X,Y,Z)  :- number(X), number(Y), Z is X*Y.
div(X,Y,Z)   :- number(X), number(Y), Z is X/Y.
minus(X,Y,Z) :- number(X), number(Y), Z is X-Y.
```

Zadanie 3. (5p) Napisz w Haskellu funkcję `ddsort`, która sortuje listę liczb całkowitych, usuwając zarazem powtarzające się liczby. Preferowane jest eleganckie rozwiązanie, w którym korzystamy z list składanych i jawnej rekursji, nie definiując żadnych funkcji pomocniczych. Można wszakże dostać 4p za poprawne rozwiązanie, nie spełniające tego warunku. Oczywiście definiując funkcję pomocniczą należy podać znaczenie jej argumentów oraz wyniku.

Matematyka dyskretna

Podaj wzór zwarty na sumę

$$\sum_{k:3|k} \binom{n}{k}$$

Algorytmy i struktury danych

Za rozwiązanie obydwu zadań z tej części można otrzymać w sumie do 9 punktów. Skala ocen: poniżej 3 punktów — ocena niedostateczna (egzamin niezdany), 3 punkty dają ocenę dostateczną, 4 — dostateczną z plusem, 5 — dobrą, 6 — dobrą z plusem, 7 albo więcej punktów daje ocenę bardzo dobrą.

Zadanie 1: przyrównywanie sekwencji nukleotydów (5 punktów)

Kiedy zostaje odkryty nowy gen, standardowe podejście do zrozumienia jego funkcji polega na przeglądaniu bazy znanych genów w poszukiwaniu bliskich odpowiedników. Bliskość dwóch genów mierzy się stopniem, w jakim mogą one zostać przyrównane. Pomyślmy więc o genach jak o słowach nad alfabetem $\Sigma = \{A, C, G, T\}$. Przyrównanie pary tkich słów polega na leksykograficznym ważonym porównaniu ciągu nukleotydów, przy czym w słowa można wstawiać tak zwane przerwy.

Rozważmy dwa przykładowe geny $x = ATGCC$ i $y = TACGCA$. Przyrównanie x i y to sposób dopasowania tych dwóch słów poprzez ich rozciągnięcie za pomocą przerw i zapisanie jednego pod drugim, na przykład:

$$\begin{array}{ccccccc} - & A & T & - & G & C & C \\ T & A & - & C & G & C & A \end{array}$$

Kreska oznacza tutaj przerwę. Litery każdego słowa muszą wystąpić po kolei. W każdej kolumnie musi wystąpić litera co najmniej jednego ze słów. Wartość przyrównania jest określana za pomocą specjalnie dobranej funkcji wagowej δ:

$$\delta : \Sigma' \times \Sigma' \mapsto \mathbb{R}$$

gdzie $\Sigma' = \{A, C, G, T, -\}$ to zbiór nukleotydów Σ powiększony o przerwę. Powyższe przyrównanie będzie więc miało następującą wartość:

$$\delta(-, A) + \delta(A, A) + \delta(T, -) + \delta(-, C) + \delta(G, G) + \delta(C, C) + \delta(C, A)$$

Skonstruuj algorytm, który dostanie na wejściu dwa geny $x \in \Sigma^n$ oraz $y \in \Sigma^m$ (dwa ciągi nukleotydów jeden o długości n i drugi o długości m) oraz funkcję wagową δ i który zwróci jako wynik przyrównanie o największej wartości. Uzasadnij poprawność opisanego algorytmu oraz oszacuj jego złożoność obliczeniową (czasową i pamięciową).

Zadanie 2: zbiory rozłączne, które najpierw łączymy a potem szukamy reprezentantów (4 punkty)

Niech dany będzie ciąg m operacji *union* i *find*, w którym wszystkie operacje *union* występują przed operacjami *find*. Udowodnij, że koszt czasowy wykonania tych operacji na drzewiastych strukturach reprezentujących zbiory rozłączne ze zbalansowanym łączeniem i kompresją ścieżek wynosi $\Theta(m)$.

Opisz budowę drzewiastej struktury danych dla zbiorów rozłącznych. Napisz jakie zadania realizują operacje *union* i *find* w tej strukturze? Na czym polega łączenie według rozmiaru/rangi oraz na czym polega kompresja ścieżki podczas wyszukiwania? Jaka jest pesymistyczna złożoność czasowa każdej z wymienionych operacji? Jaka jest złożoność czasowa wykonania ciągu m operacji *union* i *find* na zbiorach rozłącznych we wspomnianej postaci drzewiastej? Precyzyjnie uzasadnij, że złożoność czasowa wykonania takiego szczególnego przypadku ciągu operacji na zbiorach rozłącznych, w którym wszystkie łączenia zbiorów następują przed szukaniem reprezentantów jest liniowa.

Metody numeryczne

Za rozwiązanie zadań można otrzymać łącznie 12 punktów. Otrzymanie 4 pkt. gwarantuje ocenę dostateczną, próg dla dst+ to 5.5 pkt., dla db – 7 pkt., dla db+ 9 pkt., a dla bdb – 11 pkt.

1. $\boxed{\textbf{4 punkty}}$ Znajdź naturalną funkcję sklejaną trzeciego stopnia dla danych

x_k	-1	0	1
y_k	0	2	4

2. $\boxed{\textbf{4 punkty}}$ Pomiary (t_k, C_k) $(0 \le k \le N;\ t_k, C_k > 0)$ pewnej zależnej od czasu wielkości fizycznej C sugerują, że wyraża się ona wzorem

$$C(t) = \frac{1}{t^2 + A\cos t + 2}.$$

Stosując aproksymację średniokwadratową, wyznacz prawdopodobną wartość stałej A.

3. $\boxed{\textbf{4 punkty}}$ Niech dana będzie funkcja ciągła f, liczby rzeczywiste $a < b$ oraz liczba naturalna m. W języku `PWO++` procedura `RombergTable`(f, a, b, m) znajduje m-ty wiersz tablicy Romberga przybliżeń wartości całki

$$I := \int_a^b f(x)\mathrm{d}x,$$

czyli elementy $T_{0,m-1}, T_{1,m-2}, \ldots, T_{m-1,0}$ tej tablicy. Niestety, procedura ta ma **pewne ograniczenie**, mianowicie: **musi być** $m < 50$. W jaki sposób, używając procedury `RombergTable` **tylko raz** można **efektywnie** wyznaczyć wszystkie elementy 50-tego wiersza tablicy Romberga odpowiadającego całce I?

Matematyka II — Algebra

Za zadania można otrzymać 13 punktów. Aby otrzymać ocenę dostateczną, należy zdobyć 3 punkty, próg dla dst+ to 5p, dla db – 7p, dla db+ 9p, dla bdb – 11p.

Zadanie 1. (8 punktów)
Wyznaczyć wartości i wektory własne macierzy:

$$A = \begin{bmatrix} 7 & 0 & -3 \\ -9 & -2 & 3 \\ 18 & 0 & -8 \end{bmatrix}.$$

Zadanie 2. (5 punktów)
Podać współczynniki rozwinięcia wielomianu $L_3(x) = x^3 - 5x^2 + 10x - 8$ względem bazy (w bazie): $1,\ x-1,\ (x-1)(x-2),\ (x-1)(x-2)(x-3)$.