**МИНИСТЕРСТВО НА ОБРАЗОВАНИЕТО И НАУКАТА**

**У Ч Е Б Н А П Р О Г Р А М А**

**ЗА ОТРАСЛОВА ПРОФЕСИОНАЛНА ПОДГОТОВКА**

по

**УВОД В АЛГОРИТМИТЕ И СТРУКТУРИТЕ ОТ ДАННИ**

**и**

**УЧЕБНА ПРАКТИКА: УВОД В АЛГОРИТМИТЕ И СТРУКТУРИТЕ ОТ ДАННИ**

УТВЪРДЕНА СЪС ЗАПОВЕД **№ РД 09 – 5009/05.09.2017 г.**

**ПРОФЕСИОНАЛНО НАПРАВЛЕНИЕ: 481 „КОМПЮТЪРНИ НАУКИ“**

**ПРОФЕСИИ: 481010 „ПРОГРАМИСТ”**

**481020 „СИСТЕМЕН ПРОГРАМИСТ”**

**481030 „ПРИЛОЖЕН ПРОГРАМИСТ”**

**София, 2017 година**

1. **ОБЩО ПРЕДСТАВЯНЕ НА УЧЕБНАТА ПРОГРАМА**

Учебната програма по **увод в алгоритмите и структурите от данни** и по **учебна практика: увод в алгоритмите и структурите от данни** е предназначена за специалности код 4810101 „Програмно осигуряване”, код 4810201 „Системно програмиране” и код 4810301 „Приложно програмиране”.

Учебното съдържание в програмата е структурирано в три раздела, които дават възможност на учениците да получат знания, умения и компетентности за алгоритми, за анализиране и създаване на алгоритми, като се използват различни начини за описание.

1. **ЦЕЛИ НА ОБУЧЕНИЕТО ПО ПРЕДМЕТА**

Обучението по предмета има за цел учениците да придобият знания за основните видове алгоритми и професионални компетентности за решаване на задачи.

За постигане на основната цел на обучението по **увод в алгоритмите и структурите от данни** е необходимо изпълнението на следните подцели:

* придобиване на знания и умения за начините за описание и реализация на алгоритмите;
* придобиване на знания и умения за основните структури от данни и алгоритмични конструкции;
* формиране на алгоритмична култура;
* развитие на абстрактно, логическо и алгоритмично мислене.

1. **УЧЕБНО СЪДЪРЖАНИЕ**

1. Учебното съдържание е  структурирано в раздели  и теми. За всеки раздел  в програмата е определен минимален брой учебни часове. Учителят разпределя броя учебни часове за нови знания, упражнения и оценяване, при спазване изискванията за минимален брой часове по раздели.

2. Разликата между броя на учебните часове в учебния план и общия минимален брой, предвиден в учебната програма определя резерва часове. Те се разпределят по теми в началото на учебната година от учителя.

3. Раздели и теми

### Част I - Линейни структури от данни

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | Наименование на разделите | Минимален брой часове **теория** | Минимален брой часове практика |
| 1. | Въведение в алгоритмите | 2 | 2 |
| 2. | Линейни структури от данни | 6 | 6 |
| 3. | Алгоритми върху линейни структури | 4 | 4 |
|  | **Общ минимален брой часове** | **12** | **12** |
|  | **Резерв часове** | **6** | **6** |
|  | **Общ брой часове** | **18** | **18** |

### Част II - Сортиране и търсене

|  |  |  |  |
| --- | --- | --- | --- |
| № | Наименование на разделите | Минимален брой часове **теория** | Минимален брой часове практика |
| 1. | Алгоритми за сортиране | 6 | 6 |
| 2. | Алгоритми за търсене | 2 | 2 |
| 3. | Задачи върху списъци, сортиране и търсене | 4 | 4 |
|  | **Общ минимален брой часове** | **12** | **12** |
|  | **Резерв часове** | **6** | **6** |
|  | **Общ брой часове** | **18** | **18** |

### Част I - Линейни структури от данни

###### Раздел 1. Въведение в алгоритмите

1. Въведение в алгоритмите. Сложност на алгоритъм. Нотация “Big O”
2. Упражнения: изчисляване на алгоритмична сложност

**Раздел 2. Линейни структури от данни**

1. Списък и имплементации: свързан списък, разтеглив масив
2. Упражнение: имплементация на разтеглив масив
3. Упражнение: имплементация на свързан списък
4. Стекове и опашки
5. Имплементация на свързан стек
6. Имплементация на зациклена опашка

**Раздел 3. Алгоритми върху линейни структури**

1. Алгоритми върху линейни структури: подредици, нарастващи редици, площадка от еднакви елементи
2. Алгоритмични задачи върху списъци

### Част II - Сортиране и търсене

**Раздел 1. Алгоритми за сортиране**

1. Сортиране, устойчивост, бързи и бавни алгоритми, пряка селекция (Selection Sort) и имплементация
2. Метод на мехурчето (Bubble Sort) и имплементация
3. Сортиране чрез вмъкване (Insertion Sort) и имплементация
4. Сортиране чрез броене и имплементация
5. Бързо сортиране (QuickSort) и имплементация
6. Сортиране чрез сливане (MergeSort) и имплементация

**Раздел 2. Алгоритми за търсене**

1. Линейно търсене, двоично търсене, интерполационно търсене
2. Упражнения: имплементация на двоично търсене и интерполационно търсене

**Раздел 3. Задачи върху списъци, сортиране и търсене**

1. Практически задачи върху списъци, сортиране и търсене

1. **ОЧАКВАНИ РЕЗУЛТАТИ ОТ УЧЕНЕТО – ЗНАНИЯ, УМЕНИЯ И КОМПЕТЕНТНОСТИ**

В края на обучението по учебния предмет учениците придобиват компетентности за:

* основните алгоритмични конструкции;
* алгоритмите за търсене и сортиране]
* четене, разработване и описване на различни алгоритми;
* придобиване на логически и алгоритмичен подход при изпълнение на практически задания.

1. **АВТОРСКИ КОЛЕКТИВ**

Програмата е разработена от:

1. д-р Светлин Наков, СофтУни, София

Програмата е обсъдена, коригирана и оформена от експертна група към Национална програма „Обучение за ИТ кариера“ към МОН с представители на БАСКОМ, БАИТ, ИКТ клъстер и Българска аутсорсинг асоциация в състав:

1. д-р Стела Стефанова, ТУЕС към ТУ, София
2. д-р Никола Вълчанов, Програмиста, ФМИ към ПУ, Пловдив
3. Любомир Чорбаджиев, ТУЕС към ТУ, София
4. Радослав Георгиев, HackSoft, HackBulgaria, София
5. Веселина Карапеева, ОМГ "Акад. К. Попов", Пловдив
6. Ангел Георгиев, СофтУни, София
7. Ивайло Бъчваров, HackSoft, HackBulgaria, София
8. Мирослав Миронов, Мусала Софт, София
9. Владимир Начев, EPAM, София
10. Димитър Димитров, БАИТ, София
11. **ЛИТЕРАТУРА**
12. Наков С., Колев В. и колектив, Въведение в със C#, София, 2015, ISBN 978-954-400-527-6, <http://www.introprogramming.info/intro-csharp-book/>
13. Наков С. и колектив, Въведение в с Java, София, 2008, ISBN 978-954-400-055-4, <http://www.introprogramming.info/intro-java-book/>
14. Abelson H., Sussman G., Structure and Interpretation of Computer Programs, MIT Press, London, 1996