Od czego zacząć PRZYGODĘ Z PROGRAMOWANIEM

you can dev

www.youcandev-mentoring.com

SPIS TREŚCI

04

Wprowadzenie do programowania

08

Współczesna rola programowania

09

Ewolucja programowania

12

Rola programowania

15

Podstawowe koncepcje i terminologie

21

Pierwszy język programowania

24

Mity w programowaniu

29

Umiejętności programisty

you can dev

SPIS TREŚCI

43

Języki wykorzystywane na określonych stanowiskach

45

Zasoby i platformy edukacyjne

49

Mentoring

58

Książki

61

Historia sukcesu

69

Warsztat umiejętności

ZROZUMIENIE PODSTAWOWYCH KONCEPCJI I TERMINOLOGII

Podstawowe koncepcje i terminologia są kluczowe dla zrozumienia i efektywnego uczenia się programowania. Bez solidnych podstaw, nauka bardziej zaawansowanych tematów może być trudna i frustrująca. Ten podrozdział ma na celu wyjaśnienie najważniejszych pojęć i terminów, które stanowią fundament wiedzy każdego programisty.

JĘZYKI PROGRAMOWANIA

Język programowania to zestaw instrukcji używanych do tworzenia oprogramowania. Każdy język ma swoją własną składnię i zastosowanie, od prostych skryptów po kompleksowe aplikacje.

PYTHON

Język wysokiego poziomu, znany ze swojej czytelności, często używany w analizie danych i rozwoju aplikacji internetowych oraz do pracy ze sztuczną inteligencją. Bardzo często Python jest polecany jako język idealny do pierwszej styczności z kodem np. przez dzieci.

JAVA

Obiektowy język programowania, powszechnie stosowany w rozwoju aplikacji korporacyjnych i mobilnych.

JAVASCRIPT

Język skryptowy używany głównie w rozwoju części klienckiej aplikacji internetowych. Jednakże jego zastosowanie w części serwerowej dzięki używaniu dedykowanych frameworków zyskuje na popularności. Także jeżeli planujesz karierę full stack developera i chcesz pójść po relatywnie niskiej linii oporu to naucz się właśnie tego języka.

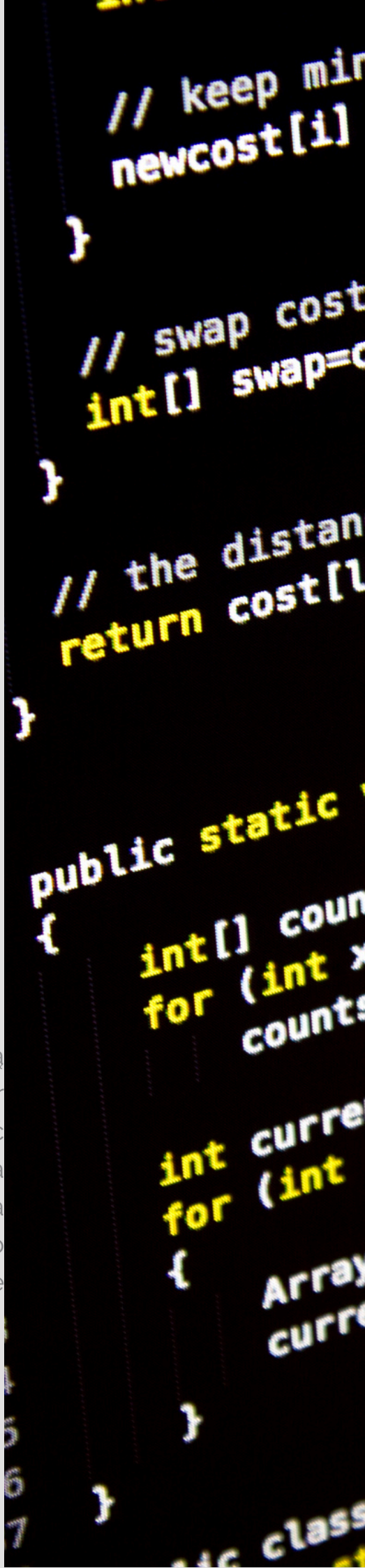
ALGORYTMY I STRUKTURY DANYCH

Algorytmy

Zestaw instrukcji służących do rozwiązania problemu lub wykonania zadania. Jest to podstawowy element programowania, który determinuje logikę i efektywność kodu. Aby przybliżyć Ci czym może być algorytm to np. przepis na naleśniki jest swego rodzaju algorytmem.
W przepisie masz listę kroków którą należy wykonać w określonej kolejności. Dopiero wtedy wyjdą naleśniki. Podobnie jest z algorytmami.

Struktury danych

Struktury danych, takie jak listy, stosy, kolejki i drzewa, są sposobami organizacji danych w programie. Wybór odpowiedniej struktury danych może znacząco wpłynąć na wydajność aplikacji. Dobry programista dobiera strukturę danych najbardziej odpowiednią do rozwiązania określonego problemu lub dobiera ją zależnie do późniejszego używania danych, które w danej strukturze są przechowywane.



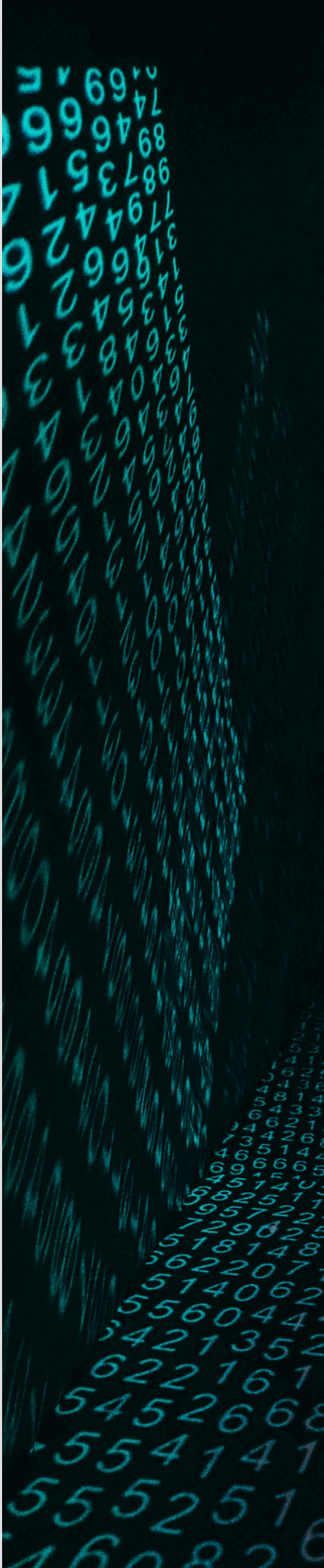
PROGRAMOWANIE OBIEKTOWE

Klasy i obiekty

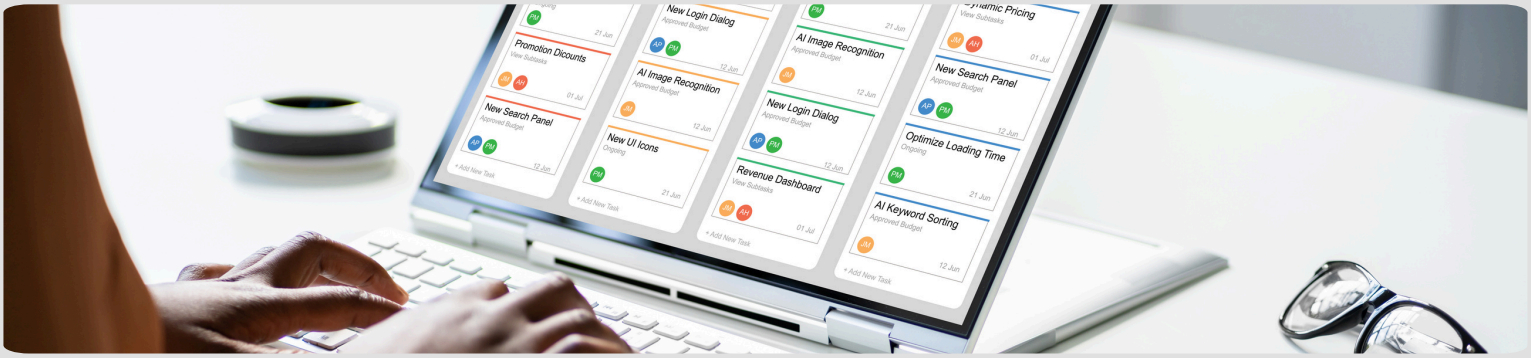
W programowaniu obiektowym, klasy są szablonami do tworzenia obiektów, które są instancjami tych klas. OOP skupia się na tworzeniu kodu, który jest modularny i można go łatwo modyfikować i rozwijać.

Dziedziczenie, Abstrakcja, Polimorfizm, Enkapsulacja

Te cztery koncepcje są fundamentem OOP. Dziedziczenie pozwala na tworzenie nowych klas na bazie istniejących, polimorfizm umożliwia traktowanie obiektów różnych klas w sposób jednolity, a enkapsulacja chroni dane wewnątrz klasy przed nieuprawnionym użyciem.



WZORCE PROJEKTOWE



Wzorce projektowe to sprawdzone rozwiązania dla często występujących problemów w projektowaniu oprogramowania. Pomagają one w organizacji kodu i jego utrzymaniu. Warto się z nimi zapoznać na późniejszym etapie nauki gdyż zrozumienie niektórych z nich nie jest prostą sprawą. Oprócz tego należy używać ich z rozwagą aby nie nadużywać wzorców, jak również przykładać dużą wagę do tego jakiego wzorca używamy. Zawsze wspominam moim podopiecznym, że w mojej opinii ważniejsze jest kiedy danego wzorca nie użyć niż kiedy go zastosować. Wynika to z faktu, że bardzo dużo młodych adeptów sztuki programowania jak już dostanie w swe ręce to bardzo użyteczne narzędzie to naturalnie wybiera swój ulubiony wzorzec starając się naginać rzeczywistość tak aby dokładnie ten wzorzec znalazł rozwiązanie w danym problemie. Nie idź tą drogą :)

DEBUGOWANIE

TESTOWANIE

Debugowanie

Debugowanie to proces identyfikacji i usuwania błędów z kodu. Jest to nieodłączny element procesu tworzenia oprogramowania.

Testowanie

Testowanie kodu polega na jego sprawdzeniu pod kątem błędów i nieprawidłowości. Wyróżniamy różne rodzaje testów, m.in jednostkowe, integracyjne i akceptacyjne.

Rozumienie podstawowych koncepcji i terminologii jest pierwszym krokiem w nauce programowania. Te fundamenty stanowią bazę, na której buduje się bardziej zaawansowana wiedza i umiejętności. Mając solidne podstawy, łatwiej jest zrozumieć złożone zagadnienia i skutecznie rozwijać swoje umiejętności programistyczne. Zapoznanie się z tymi podstawowymi pojęciami ułatwi każdemu początkującemu programiście rozpoczęcie swojej przygody z kodowaniem.

WARSZTAT UMIEJĘTNOŚCI

W tym rozdziale znajdziesz materiały dodatkowe dołączone do tego ebooka:

- Test predyspozycji
- Ściągawka do rozwiązywania problemów w kodzie
- Rób & Nie rób - lista rzeczy które należy robić lub których unikać podczas nauki programowania
- Metoda 4 kroków, która pomoże Ci rozwiązywać problemy programistyczne
- Programistyczne BINGO
- Znajdź ukryte słowa

Rozwiązywanie problemów w kodzie

ŚCIĄGAWKA

Nie rozumiesz całości kodu

Rozbij program na mniejsze części i próbuj zrozumieć kawałek po kawałku. Gdy już zrozumiesz pierwszy kawałek zabierz się za kolejny. Wykonuj takie iteracje aż do końca.

Nie masz pomysłu jak napisać kod

Użyj pseudokodu. Rozpisz sobie na kartce własnymi słowami co się ma po kolei stać w programie aby osiągnąć zamierzony cel.

Nie masz pomysłu na rozwiązanie problemu

Użyj metody gumowej kaczki. Wytłumacz jej problem z jakim się mierzysz a rozwiązanie samo wpadnie do Twojej głowy.

Długo myślisz nad rozwiązaniem i pomysłu nadal brak

Odpocznij od programowania jakiś czas. Nie zalecamy zbyt długiego odpoczynku ale odpoczynek rzędu kilku godzin nie zaszkodzi.

Wydaje Ci się że ktoś już musiał kiedyś rozwiązać ten problem

Poszukaj rozwiązania tego problemu w internecie. Bardzo możliwe że ktoś rozwiązał ten problem przed Tobą i będziesz mógł zainspirować się tym rozwiązaniem w swoim projekcie.

Długo pracujesz nad konkretnym rozwiązaniem konkretnego problemu

Wyrzuć cały kod, który został napisany dotychczas aby ten jeden konkrenty problem rozwiązać i spróbuj napisać go od nowa. Lecz nie wykonuj pracy odtwórczej. Pomyśl nad inną koncepcją. Najważniejsze to pozbyć się starego kodu.

Nie rozumiesz działania pojedynczej linii kodu

Sprawdź w dokumentacji wszystkie funkcję, które użyte są w tej linii kodu. I kawałek po kawałku staraj się połączyć ich użycie w spójną całość. Korzystaj z oficjalnej dokumentacji narzędzia, którego używasz.

Nie wiesz dlaczego Twoja zmienna ma taką wartość a nie inną

Użyj debbugera, to bardzo potężne i pomocne narzędzie, które pozwoli Ci sprawdzić jakie wartości są przechowywane w Twoich zmiennych podczas działania programu i wiele innych.

BINGO

Jak grać:

1. Zaczynasz z pustą kartą BINGO.
2. Celem jest wykonanie zadań z różnych kategorii, aby osiągnąć bingo. Bingo można osiągnąć poprzez ukończenie zadań w linii poziomej, pionowej lub na ukos.
3. Po wykonaniu danego zadania zaznaczasz je w karcie BINGO.
4. Pierwsza osoba, która osiągnie bingo (i potwierdzi wykonanie zadań, np. poprzez zrzuty ekranu, linki do projektów, itp.), wygrywa nagrodę niespodziankę od firmy you can dev.

Zainstaluj swoje pierwsze IDEA	Napisz swój pierwszy program	Poznaj podstawowe typy danych	Uczestnicz w webinarze o programowaniu	Znajdź mentora
Ukończ kurs dot. programowania	Stwórz stronę internetową w HTML/CSS	Rozwiąż 5 problemów typu codility	Naucz się podstaw GITa	Przeczytaj artykuł o algorytmach
Obejrzyj tutorial na YouTube o Javie	Zainstaluj i skonfiguruj IDEA	Wolne pole	Zrozum różnicę między Java a JS	Zapisz się na programistyczny newsletter
Zaprojektuj swoją pierwszą aplikację mobilną	Naucz się podstaw SQL	Zapoznaj się z możliwymi atakami na aplikację webową	Weź udział w hackatonie	Przeczytaj biografię znanego programisty
Stwórz repozytorium na GitHubie	Zrozum podstawy OOP	Zainstaluj dodatek do przeglądarki	Weź udział w konkursie programistycznym	Napisz automatyzację do 1 codziennej czynności

DZIĘKUJEMY ŻE JESTEŚ Z NAMI !

Jeśli ten fragment rozbudził Twoją pasję do programowania, nie przegap okazji, by odkryć pełen potencjał z naszą kompletną wersją eBooka. Znajdziesz ją w naszym sklepie – zacznij swoją programistyczną przygodę już dziś!

you can dev

> “Jeśli chcesz gdzieś dojść, najlepiej znajdź kogoś, kto już tam doszedł.”

Robert Kiyosaki