The background of the slide features the coat of arms of the University of Warsaw, which depicts a white eagle with its wings spread, perched on a golden crown topped with a six-pointed star. The eagle is surrounded by oak leaves and acorns. The entire scene is set against a dark blue, slightly textured background.

Krzysztof Diks
Instytut Informatyki
Uniwersytet Warszawski

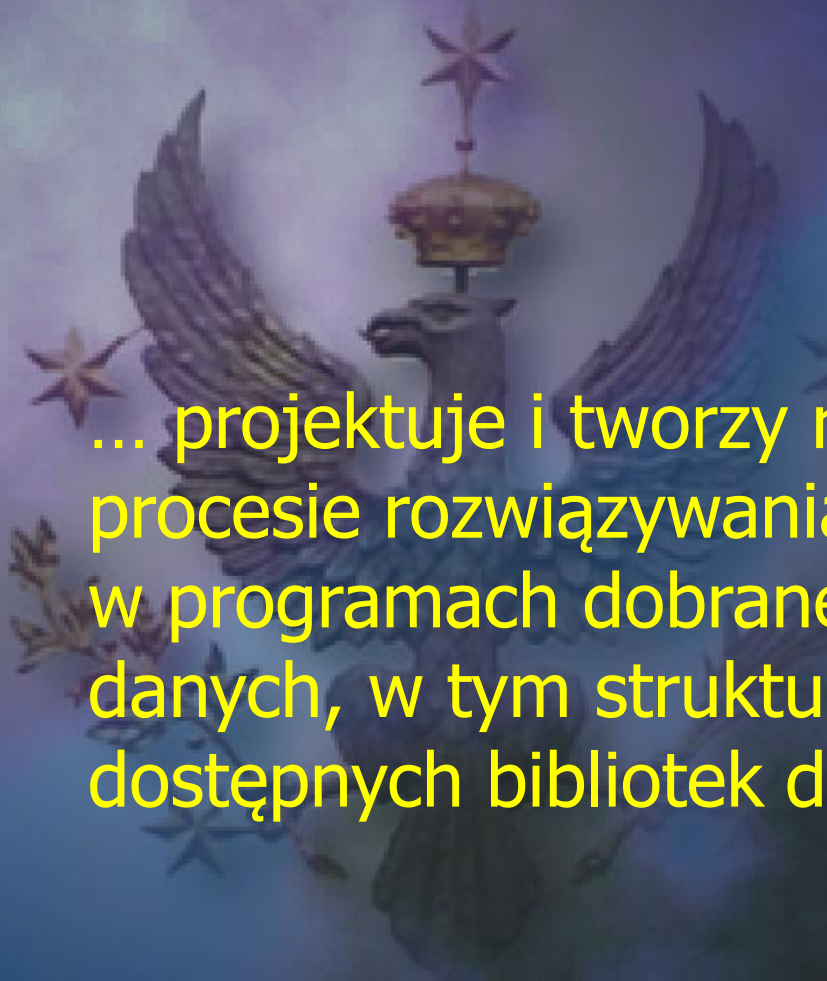
**Algorytmika i programowanie w
nowej podstawie programowej
nauczania informatyki**

The background of the slide features a faint, light-colored watermark of the Polish coat of arms, which includes a white eagle with its wings spread, perched on a crown, set against a dark blue background.

INFORMATYKA

ZAKRES PODSTAWOWY I ROZSZERZONY

- I. Rozumienie, analizowanie i rozwiązywanie problemów na bazie logicznego i abstrakcyjnego myślenia, myślenia algorytmicznego i sposobów reprezentowania informacji.
- II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera oraz innych urządzeń cyfrowych: układanie i programowanie algorytmów, organizowanie, wyszukiwanie i udostępnianie informacji, posługiwanie się aplikacjami komputerowymi.

The background of the slide features a faint, stylized version of the Polish coat of arms, which is a white eagle with its wings spread, holding a golden crown on its chest. The eagle is set against a light blue and white background, which is partially obscured by the dark blue gradient of the slide.

... projektuje i tworzy rozbudowane programy w procesie rozwiązywania problemów, wykorzystuje w programach dobrane do algorytmów struktury danych, w tym struktury dynamiczne i korzysta z dostępnych bibliotek dla tych struktur;

David Harel: **ALGORITHMICS**. *The Spirit of Computing*. (Rzecz o istocie informatyki. Algorytmika.)

*Przepisy określiliśmy mianem algorytmów, obszar zaś ludzkich dociekań, wiedzy i doświadczeń dotyczących algorytmów nazwiemy **algorytmiką**.*

...Algorytmika to więcej niż dział informatyki. Tkwi ona w centrum wszystkich działów informatyki ...



Donald E. Knuth:

Proces przygotowywania programu dla komputera cyfrowego jest pociągający nie tylko ze względu na potencjalne korzyści ekonomiczne czy naukowe. Wiązą się z nim również przeżycia estetyczne, podobne do tworzenia poezji lub komponowania muzyki.



ważne własności algorytmów:

❑ poprawność

❑ złożoność obliczeniowa



Przykład: zadanie maturalne z roku 2005

Dane: tablica liczb całkowitych $a[1..n]$

Wynik: najlepsza = $\max(\{0\} \cup \{s[i,j]: 1 \leq i \leq j \leq n\})$,
gdzie $s[i,j] = a[i] + a[i+1] + \dots + a[j]$.

Przykład

$a = [2, 3, -2, 3, -5, -1, 2, -1, 3, -2, 4, -1, 2, -1]$

najlepsza = 7

Rozwiązanie 1

Obliczamy sumy $s[i,j]$ wprost z definicji i wybieramy największą z nich (0, gdy wszystkie liczby w tablicy a są mniejsze od 0).

```
najlepsza = 0;
for (i = 1; i <= n; i++)
    for (j = i; j <= n; j++){
        s = 0;
        for (k = i; k <= j; k++)
            s = s + a[k]; // operacja dominująca
        if (s > najlepsza) najlepsza = s;
    }
```

Liczba operacji dominujących:

$$T_1(n) = n + 2(n-1) + 3(n-2) + \dots + n = \mathbf{n(n+1)(n+2)/6}.$$

n	100	1000	10,000	100,000
$T_1(n)$	171,700	167,167,000	166,716,670,000	166,671,666,700,000

Rozwiązanie 2

Korzystamy z obserwacji, że $s[i,j+1] = s[i,j] + a[j+1]$.

```
najlepsza = 0;
for (i = 1; i <= n; i++){
    s = 0;
    for (j = i; j <= n; j++){
        s = s + a[j]; // operacja dominująca
        if (s > najlepsza) najlepsza = s;
    }
}
```

Liczba operacji dominujących jest równa liczbie przedziałów, czyli

$$T_2(n) = n*(n+1)/2$$

n	100	1000	10,000	100,000
$T_1(n)$	171,700	167,167,000	166,716,670,000	166,671,666,700,000
$T_2(n)$	5,050	500,500	50,005,000	5,000,050,000

Rozwiązanie 3

Rozumowanie „indukcyjne”.

W jaki sposób z rozwiązania dla tablicy $a[1..i-1]$ dostać rozwiązanie dla $a[1..i]$?

Niezmiennik:

- *najlepsza* jest najlepszą sumą w $a[1..i-1]$
- *naj_prawa* jest najlepszą sumą w $a[1..i-1]$ dla przedziałów o prawym końcu w $i-1$.

```
najlepsza = 0; naj_prawa = 0;
for (i = 1; i <= n; i++){
    naj_prawa = max(0, naj_prawa + a[i]);
    if (naj_prawa > najlepsza)
        najlepsza = naj_prawa;
}
```

Złożoności obliczeniowe poszczególnych algorytmów

n	100	1000	10,000	100,000
$T_1(n)$	171,700	167,167,000	166,716,670,000	166,671,666,700,000
$n(n+1)/2$	4,950	500,500	50,005,000	5,000,050,000
n	100	1,000	10,000	100,000

PRZYGODY BAJTAZARA

25 LAT OLIMPIADY INFORMATYCZNEJ

WYBÓR ZADAŃ



 PWN



KRZYSZTOF DIKS

Krzysztof Diks jest profesorem w Instytucie Informatyki Uniwersytetu Warszawskiego. Specjalizuje się w algorytmice, wykłada *Algorytmy i struktury danych*. Od 1994 roku związany jest z Olimpiadą Informatyczną; od 1999 roku jest przewodniczącym Komitetu Głównego Olimpiady. W 2005 roku był przewodniczącym Komitetu Organizacyjnego Międzynarodowej Olimpiady Informatycznej, która odbyła się w Polsce. Jest pomysłodawcą i współorganizatorem Potyczek Algorytmicznych - najpopularniejszego konkursu algorytmicznego w Polsce. Jest autorem wielu zadań konkursowych i popularyzatorem informatyki. Od 1999 roku jest, wspólnie z prof. Janem Madeyem, opiekunem reprezentacji Uniwersytetu Warszawskiego w konkursach programistycznych. Był współopiekunem mistrzów świata w programowaniu zespołowym ACM ICPC w latach 2003 i 2007.

Dziękuję!