#### Description de l’algorithme

#### Caractérisation des cas

#### Mesure de l’efficacité du pire des cas

#### Mesure de l’efficacité du meilleur des cas

#### Mesure de l’efficacité moyenne

#### Caractérisation du cas général

//////////////////////////////////////////////////

MESURE DE L EFFICACITÉ

EXEMPLE 1 (page 4)

#### Description de l’algorithme

l’algorithme calcul la somme entre 2 entiers

#### Caractérisation des cas

un seul cas est présent, représentant l’ensemble des cas

#### Mesure de l’efficacité du pire des cas

#### Mesure de l’efficacité du meilleur des cas

#### Mesure de l’efficacité moyenne

#### Caractérisation du cas général

//////////////////////////////////////////////////

EXEMPLE 2 (page 4)

#### Description de l’algorithme

L’algorithme retourne vrai ou faux suivant la valeur de a

Si a<10, retourne vrai, si a>8, retourne vrai, sinon retourne faux

#### Caractérisation des cas

Le pire des cas est lorsque a<=10

Le meilleur des cas est lorsque a>10

#### Mesure de l’efficacité du pire des cas

#### Mesure de l’efficacité du meilleur des cas

#### Mesure de l’efficacité moyenne

#### Caractérisation du cas général

//////////////////////////////////////////////////

EXEMPLE 3 (page 5)

#### Description de l’algorithme

Cherche la valeur a dans le tableau, retourne vrai si il le trouve, sinon retourne faux

#### Caractérisation des cas

Le pire cas est lorsque la valeur du paramètre « A » n’est pas dans le tableau

Le meilleur cas est lorsque la valeur du paramètre « A » est dans l’index 0 du tableau

#### Mesure de l’efficacité du pire des cas

#### Mesure de l’efficacité du meilleur des cas

#### Mesure de l’efficacité moyenne

1 chance sur 2 de faire le meilleur des cas, 1 chance sur 2 de faire le pire des cas(c’est faux mais on simplifie pour l’exercice)

//////////////////////////////////////////////////

MESURE D EFFICACITÉ

EXERCICE 1(page6)

#### Description de l’algorithme

La fonction cherche dans le tableau la plus grande valeur et l’affecte à la variable « VAL »

(aide = N représente la taille du tableau. Nous, nous voulons travailler avec l’indice pour pouvoir faire tab[i]. Il faut donc transformer la valeur maximum de la taille de tableau en indice maximum, sois N -1

Aide= On test chaque valeur du tableau sauf celle à l’index 0. Notre nombre de tests à faire représente donc tous les index du tableau – l’index0. Voilà pourquoi « i » = 1 pour commencer directement sur l’index1.

#### Caractérisation des cas

Le pire des cas est lorsque le tableau est rangé avec des valeurs croissantes

Le meilleur des cas est lorsque la plus grande valeur est à l’index 0 du tableau.

#### Mesure de l’efficacité du pire des cas

#### Mesure de l’efficacité du meilleur des cas

#### Mesure de l’efficacité moyenne

1 chance sur 2 de faire le meilleur des cas, 1 chance sur 2 de faire le pire des cas(c’est faux mais on simplifie pour l’exercice)

#### Caractérisation du cas général

• 1er cas, la valeur maximum est à l’indice 1

• 2ème cas, 1 chance sur 2 d’avoir valeur de l’index1 > valeur de l’index0 et la valeur maximum du tableau est à l’index 2

On peut voir qu’entre le cas n1 représentant la valeur max sur l’index1 et le cas numéro2 représentant la valeur max sur l’index2, il y a eu + au résultat.

Il y a donc récurrence et il suffit donc de savoir la position de la valeur maximum pour connaître le résultat.

Le ne s’additionne qu’1 fois à partir de l’index2, 2fois à partir de l’index3, etc.

Pour transformer le positionvaleurmax en multiplicateur, il faut donc faire -1 sur cette valeur.

//////////////////////////////////////////////////

EXERCICE 2

(on fait n-1 car j représente les index du tableau. Le dernier index du tableau représente donc tailledutableau-1)

#### Description de l’algorithme

Affiche les valeurs du tableau supérieur à « A » et qui sont entre l’index « i » et la fin du tableau

#### Caractérisation des cas

Le pire cas est lorsque chaque valeur du tableau entre l’index « i » et la fin du tableau est supérieur à la variable « A ».

Le meilleur cas est lorsque chaque valeur du tableau entre l’index « i » et la fin du tableau est inférieur à la variable « A ».

Le cas moyen est lorsque chaque valeur du tableau entre l’index « i » et la fin du tableau est inférieur ou supérieur à la variable « A » de manière égale.

#### Mesure de l’efficacité du pire des cas ( !inverser dans la leçon)

#### Mesure de l’efficacité du meilleur des cas ( !inverser dans la leçon)

#### Mesure de l’efficacité moyenne

/////////////////////////////////////////////////////

/////////////////////////////////////////////////////

Exemple appel itératif (bas de page 10)

//////////////////////////////////////////////

/////////////////////////////////////////////

MESURE DE COMPLEXITÉ

EXERCICE 1(page 12)

#### Description de l’algorithme

Pour chaque valeur du tableau, si la valeur enregistrer est inférieur à la valeur itéré dans le tableau, alors la nouvelle valeur enregistrer devient la valeur itéré.

#### Caractérisation des cas

Le pire cas est lorsque le tableau est rangé de manière croissant.

Le meilleur cas est lorsque l’index0 du tableau possède la valeur la plus grande

Le cas moyen est si les valeurs sont 1fois sur 2 plus grande et 1fois sur 2 plus petit

#### Mesure de la complexité

#### Mesure de la complexité de g1() dans le meilleur cas

#### Mesure de la complexité de g1() dans le pire cas

#### Mesure de la complexité g1() dans le cas moyen

Le cas moyen est la moyenne entre le pire et le meilleur des cas

#### Mesure de la complexité de f1() dans le meilleur cas

#### Mesure de la complexité de f1() dans le pire cas

#### Mesure de la complexité f1() dans le cas moyen

/////////////////////////////////////////////

EXERCICE 2(page14)

#### Description de l’algorithme

Affiche toutes les valeurs supérieures à A de l’index « i » jusqu’à l’index à la fin du tableau

#### Caractérisation des cas

Le pire cas est lorsque toutes les valeurs d’index sont supérieures à A

Le meilleur cas est lorsque toutes les valeurs d’index sont inférieures à A

Le cas moyen est lorsque toutes les valeurs d’index sont moitié inférieures moitié supérieures à A

#### Mesure de la complexité

#### Mesure de la complexité de g1() dans le meilleur cas

#### Mesure de la complexité de g1() dans le pire cas

#### Mesure de la complexité g1() dans le cas moyen

#### Mesure de la complexité de f1() dans le meilleur cas

#### Mesure de la complexité de f1() dans le pire cas

#### Mesure de la complexité f1() dans le cas moyen

/////////////////////////////////////////////

EXERCICE 3(page15)

#### Description de l’algorithme

Retourne la plus grande valeur du tableau entre l’index de J et l’index de la fin du tableau

#### Caractérisation des cas

Le pire cas est lorsque toutes les valeurs d’index sont rangées de manière chronologique

Le meilleur cas est lorsque la valeur de l’index J est la plus grande valeur du tableau

Le cas moyen est lorsque toutes les valeurs d’index sont moitié inférieures moitié supérieures à J

#### Mesure de la complexité

FAUX NON TERMINÉ

#### Mesure de la complexité de g1() dans le meilleur cas

#### Mesure de la complexité de g1() dans le pire cas

#### Mesure de la complexité g1() dans le cas moyen

#### Mesure de la complexité de f1() dans le meilleur cas

#### Mesure de la complexité de f1() dans le pire cas

#### Mesure de la complexité f1() dans le cas moyen